DYNABEE: DYNAMICAL SYSTEMS FOR MODELING HONEY BEE COLONY BEHAVIOR

Anonymous authors
Paper under double-blind review

ABSTRACT

Given sparse and noisy measurements, how can we model, forecast, understand the colony behavior? Domain experts strictly demand that the method is explainable, fast, and accurate.

We propose DYNABEE, which consists of a carefully designed dynamical system of differential equations, along with their stability analysis. We developed the equations from first principles, in close collaboration with entomologists. As a result, DYNABEE has only few, perfectly interpretable parameters to fit (foraging rate, mortality rate, etc).

Thus, DYNABEE has the following desirable properties: (a) Accurate: experiments on the largest available dataset from real hives show good accuracy; (b) Explainable: our model parameters have physical meaning; (c) Scalable: DYNABEE is linear on the input size; (d) Transferable: the fitted parameter values give good forecasting for new bee colonies, too.

1 INTRODUCTION

Given a colony's sparse, noisy inspection records, how can we forecast its future status in actionable quantities (the no of honey, larvae, and bee frames, plus the mite level)? The ideal method should be accurate, and of course, explainable to the beekeepers, who make the final call on treatment.

Goal: The long-term goal of our research is to use the records beekeepers already collect to anticipate colony stress and health issues early, and issue warnings and suggest remedies. In this paper, we take the first step: forecasting the next inspection from a colony's past manual inspection records.

Why study bees: Without honeybees, agriculture will suffer. Honeybees are vital pollinators for about a third of what we eat (Eilers et al., 2011). Bee pollination contributes an estimated $550 billion globally, and $29 billion in the U.S. (Potts et al., 2016). With beekeepers losing almost half of their colonies in 2023 (Steinhauer et al., 2024) due to parasites, pesticides, and climate issues (Brunet & Fragoso, 2024) , it became crucial to understand and forecast colony behavior.

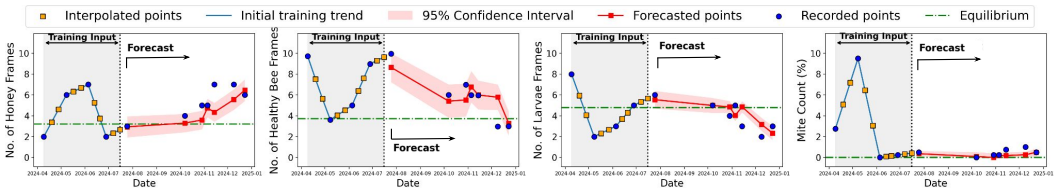
Why inspection records, not sensors: Low-cost sensing and IoT enable hive monitoring through temperature, humidity, weight, and sound signals (Davidson et al., 2020; Hossain et al., 2024). However, sensor deployment is costly, and raw sensor traces do not tell a beekeeper what to do. Most apiaries do not have them and may not afford to install them on every hive. Beekeepers actually act on the inspection quantities, like honey, brood, and bee frame counts, and the mite count (Honey Bee Health Coalition, 2022). Thus, these numbers are the most decision-relevant signal to analyze.

Explainability is a must: Inspection records are inherently sparse. Hive inspection is laborious and disrupts the colony, so the beekeepers cannot do it more often than every 2-4 weeks . Forecasting the next inspection would therefore let them anticipate problems and act between visits. Standard forecasting methods, whether linear or deep, make this hard with no biological interpretation (Section 2). Thus, explainability is a show-stopper for adoption, as per practitioners.

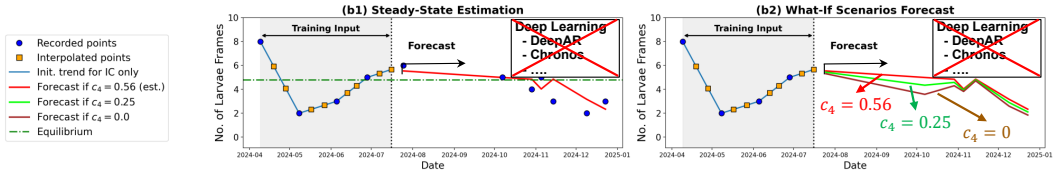
Proposed solution: DYNABEE. We propose a 'white box' dynamical system of differential equations, developed from first principles in collaboration with entomologists. DYNABEE fits historical inspection records, recovers hidden states, and forecasts future colony dynamics. Its equilibria and stability allow experts to assess the biological plausibility of the fitted model. The contributions are:

- Accurate: Applied on real inspection data, DYNABEE does good forecasting (see Figure 1a).
- Explainable: DYNABEE consists of differential equations, derived from discussions with entomologists; every parameter has physical meaning (honey-bee death rate, etc), so experts can check both the parameters and the predicted equilibria and what-if scenarios (see Figure 1b).

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) DYNABEE is accurate



(b) Power of our white-box method: DYNABEE is explainable, and more

Figure 1: (a) the one-step-ahead forecasting (red line) is close to the actual (blue dots), falling in 95% confidence interval in most cases. The equilibrium points of DYNABEE (green lines) are stable and agree with expert intuition; (b) DYNABEE can do tasks that **no** black-box/deep-learning method can do: (b1) estimation of equilibrium point (dashed line) (b2) what-if-scenarios: the higher the queen egg-laying rate c_4, the higher the larvae population.

- Scalable: DYNABEE is linear on the input size; it takes from seconds to minutes on a stock laptop, to process several months' worth of data.
- Transferable: The fitted parameters carry over across colonies; using the parameters found on one set of hives, DYNABEE can still forecast well, applied on different colonies (see Section 5.2).

Reproducibility: Our data and source code is available on `https://anonymous.4open.science/r/DynaBee-26EC/`.

2 RELATED WORK

There is a lot of related work in CS ((a) and (b) below) and biology ((c), (d) and (e)), but only DYNABEE matches all the desirable features (see Table 1).

Property \ Method	DYNABEE	(a) Trad. Linear Forec. (Yi et al., 1998)	(b) Deep L. Forec. (Lim et al., 2021)	(c) Agent-Based Bee Models (Becher et al., 2014)	(d) ODE-Based Bee Models (Khoury et al., 2013)	(e) Bee App. (Wang et al., 2024)
first principles	✓			✓	✓	✓
parsimony	✓	✓			?	
white-box	✓	✓		?	?	?
interpretable parameters	✓			✓	✓	?
scalable	✓	✓	?	?	?	?
verified on honeybee data	✓	?	?			

Table 1: DYNABEE wins on features, fulfilling all desired properties. Columns (a)–(e) correspond to the related-work groups in Section 2. Here '?' means 'maybe, depending on the implementation'.

(a) Traditional time series forecasting. Time series mining have been extensively studied (Yeh et al., 2016; Matsubara et al., 2014). Classical linear approaches include autoregressive models (Box et al., 2015), such as ARIMA, and its variants, and exponential smoothing methods such as Holt-Winters (Hamilton, 2020). However, they lack biologically interpretable parameters and cannot capture nonlinear colony dynamics. Thus, even our simplest model is non-linear (Sec. 3.1).

(b) Deep learning forecasters. Modern approaches include DeepAR (Salinas et al., 2020), Autogluon-Timeseries (Shchur et al., 2023) and Chronos (Ansari et al., 2024; 2025). They lack interpretable parameters, they can not estimate steady-state values, nor do what-if scenarios (Fig 1b).

Symbol	Definition
E	# eggs (in frames)
Q	0/1 (whether queen exists)
L_H	# healthy larvae (in frames)
L_M	# mite-infected larvae (in frames)
L	$= L_H + L_M$ total # larvae (in frames)
B_H	# healthy adult bees (in frames)
B_M	# mite-infected adult bees (in frames)
B	$= B_H + B_M$ total # adult bee frames
D	# dead bee (in frames)
H	# honey frames
u_M	0/1 (mite treatment event)

(a)

Symbol	Definition	Range (per day)	DYNABEE est.
c_1	foraging rate per bee frame	$0 \leq c_1 \leq 0.1$ honey frames	0.073
c_2	honey consumption rate per (adult) bee frame	0.0122 honey frames	0.0122^*
c_3	honey consumption rate per brood frame	$0 \leq c_3 \leq 0.0342$ honey frames	0.0244^*
c_4	egg laying rate of queen	$0 \leq c_4 \leq 0.571$ honey frames	0.561
c_5	development rate of egg population into larvae	0.33	0.33^*
c_6	emergence rate of larvae	0.048	0.048^*
c_7	death rate of larvae	$0.001 \leq c_7 \leq 0.05$	0.0042
c_7'	death rate of larvae (starvation)	$0.1 \leq c_7' \leq 0.24$	0.0084^*
c_8	death rate of bees	$0.01 \leq c_8 \leq 0.08$	0.012
c_8'	death of bees due to starvation	$0.1 \leq c_8' \leq 0.24$	0.024^*
c_9	death rate of mite-infested bees	$0.03 \leq c_9 \leq 0.5$	0.057
c_9'	death rate of mite-infested larvae	$0.03 \leq c_9' \leq 0.5$	0.133
c_{10}	rate of healing from mites with treatment	$0 \leq c_{10} \leq 1$	0.027

(b)

Table 2: (a) Symbols & Definitions (b) List of model coefficients; definitions and estimated values collected from the literature Rodney & Purdy (2020); Rueppell et al. (2007). * We use fixed c_2, c_4, and c_5 from the literature and $c_3 = 2c_2$, $c_7' = 2c_7$, $c_8' = 2c_8$ (Deatils in Sec. 5).

(c) Agent-Based Bee Models. Among the most popular models, BEEHAVE (Becher et al., 2014) is an agent-based simulator that requires dozens of parameters, proposes no differential equations, and only qualitative validation against literature. Bee++ (Betti et al., 2017) is a hybrid agent-based simulation with a differential equation. However, it was never validated in real-world.

(d) ODE-Based Bee Models. Khoury et al. (Khoury et al., 2013) use a compact ODE model but make assumptions such as a negligible death rate for hive bees, omit mites and treatments, do not separate larvae from eggs, and validate only against a single aggregate endpoint.

(e) Bee Applications. Two studies (Agatz et al., 2019; Wang et al., 2024) place BEEHAVE output beside real measurements without statistical comparison. Other works (Schödl et al., 2022; Groeneveld et al., 2024) use existing models on simulated scenarios, also without statistical validation.

3 PROPOSED MODEL: DYNABEE

In this section, we present our modeling framework. We begin with DYNABEE$_0$, which models honey and healthy adult bee dynamics (Sec. 3.1), extend it to DYNABEE$_{mites}$ with *Varroa* mite infestation (Sec. 3.2), and finally introduce DYNABEE, our full model incorporating larvae dynamics (Sec. 3.3). Tables 2a and 2b summarize the model symbols and coefficients, respectively. Capital symbols denote time series (e.g., $H = H(t)$), with t omitted for brevity, and dots denote time derivatives (e.g., $\dot{H} = \partial H/\partial t$). Figure 2a summarizes the throttling factors used in the paper. Background on dynamical systems is provided in Appendix A.1.

3.1 BASE MODEL - DYNABEE$_0$

We begin with the simplest case: eggs laid by the queen (Q) develop directly into adult bees (B), without an explicit larval stage or *Varroa* mite infestation.

Model 1 (DYNABEE$_0$)**.** *Our base model is governed by the following equations:*

$$\dot{H} = \overbrace{c_1 B p(H)}^{\textit{foraging}} - \overbrace{c_2 B e(H)}^{\textit{consumption}} \tag{1}$$

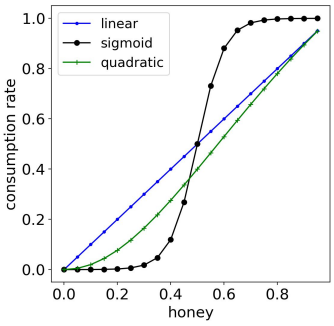
$$\dot{B} = \underbrace{c_4 Q p(H) e(H)}_{\textit{birth}} - \underbrace{f(H) B}_{\textit{death}} \tag{2}$$

Honey accumulation depends on foraging and consumption, both regulated by honey availability. Bees stop foraging when the hive is full ($H = H_M$) and reduce consumption when honey is scarce.

Function	Linear	Sigmoid	Quadratic (≈Michaelis-Menten)
$p(H)^\dagger$	$1 - \frac{H'}{H_M}$	$\frac{1}{1+e^{-k_p(H_M - H')}}$	$\frac{H_M^2}{H_M^2 + H'^2}$
$e(H)$	$\frac{H}{H_M}$	$\frac{1}{1+e^{-k_e(H-H_0)}}$	$\frac{H^2}{H^2+H_0^2}$
$g(H)$	$c_7 e(H) + c_7' p(H)$	$c_7' + (c_7 - c_7')e(H)$	$c_7' + (c_7 - c_7')e(H)$
$g_M(H)$	$c_9' e(H) + c_7' p(H)$	$c_7' + (c_9' - c_7')e(H)$	$c_7' + (c_9' - c_7')e(H)$
$f(H)$	$c_8 e(H) + c_8' p(H)$	$c_8' + (c_8 - c_8')e(H)$	$c_8' + (c_8 - c_8')e(H)$
$f_M(H)$	$c_9 e(H) + c_8' p(H)$	$c_8' + (c_9 - c_8')e(H)$	$c_8' + (c_9 - c_8')e(H)$

† For DYNABEE, $H' = H + L + E$. For DYNABEE$_0$, DYNABEE$_{mites}$ and starvation interpolation functions , $H' = H$.

(a)



(b)

Figure 2: (a) Definition of different throttling functions; (b) Plots of candidate 'starvation' throttling functions. 'Abundance' and 'interpolation' functions are similar in spirit.

We capture these effects using the throttling functions $p(H)$ and $e(H)$. Similarly, bee growth depends on queen presence (Q), egg-laying rate c_4, and honey-dependent throttling factors that regulate egg laying. Bee mortality also increases when honey is scarce.

Next, we define the abundance throttling function $p(H)$, starvation throttling function $e(H)$, and starvation interpolation function $f(H)$. Figure 2b gives the intuition for 'starvation' throttling.

Definition 1. *(Abundance Throttling Function, $p(H)$) A hive has a fixed number of frames, H_M, available for honey storage or egg laying. When $H = H_M$, no space remains, so foraging and egg laying stop, i.e., $p(H) = 0$. For $H \in [0, H_M)$, space remains available for both activities.*

Definition 2. *(Starvation Throttling Function, $e(H)$) As the honey reserve becomes critically low, i.e., $H \approx 0$, bees experience starvation, and the queen stops egg laying. When $H \in [0, H_M)$, the colony has sufficient food resources to sustain normal hive activity.*

Definition 3. *(Starvation Interpolation Function, $f(H)$) Bee mortality depends on food availability: it remains low with sufficient honey reserves but increases as honey falls below a critical threshold, inducing starvation.*

We present these throttling functions in Figure 2a using three alternative functions: (i) Linear, (ii) Sigmoid, and (iii) Quadratic (Michaelis-Menten).

3.1.1 EQUILIBRIUM POINTS AND STABILITY ANALYSIS.

Now, we derive the steady-state solution of DYNABEE$_0$. For simplicity, we derive the linear case; the same procedure applies to sigmoid and quadratic formulations, but with less tractable expressions.

Lemma 1. *For strictly positive model parameters, $c_i > 0, \forall i$, any biologically feasible equilibrium*

$$(H^*, B^*) = \left(\frac{c_1}{c_1 + c_2} H_M, \frac{c_4 Q\, p(H^*) e(H^*)}{f(H^*)} \right)$$

corresponding to a living colony is stable, where,

$$p(H^*) = \frac{c_2}{c_1 + c_2}, e(H^*) = \frac{c_1}{c_1 + c_2} \text{ and}$$

$$f(H^*) = c_8 \frac{c_1}{c_1 + c_2} + c_8' \frac{c_2}{c_1 + c_2}$$

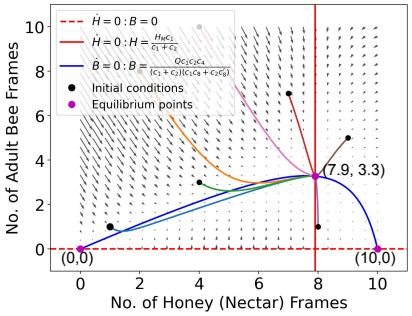


Figure 3: DYNABEE$_0$ is explainable: all initial conditions (black circles) converge to a reasonable, stable equilibrium point (purple circle).

Discussion. Due to space constraints, we provide the proof in Appendix A.2. For a fixed set of parameters ($c_1 = 0.046, c_2 = 0.0122, c_4 = 0.571, c_8 = 0.01, c_8' = 2c_8$) and in the absence of external disturbances, any living colony converges to the stable equilibrium point $(H^*, B^*) =$

$(7.9, 3.3)$ (Figure 3). This equilibrium is consistent with domain experts' expectations, in which both honey reserves and bee population remain sustainable.

Lemma 2. *For strictly positive model parameters, $c_i > 0, \forall i$, any biologically feasible equilibrium $(H_d, 0)$ corresponding to a dead colony is a saddle point for any $H_d \in [0, H_M]$.*

Discussion. As shown in Figure 3(a), all initial conditions (black dots) are repelled by the saddle equilibria $(H^*, B^*) = (0, 0)$ and $(10, 0)$, indicating their instability. Hives with zero bees will never recover, even if it has stored honey.

3.2 With Mites - DynaBee$_{mites}$

Now, we introduce an essential concept, *mite infestation*: Mite-infested adult bees (B_M), can infest cells within the hive frames. If the queen (Q) lays eggs in mite-infested cells, the emerging adult bees will become part of the mite-infested bee population, B_M. Infected adult bees may recover if treatment is applied (u_M).

Model 2 (DynaBee$_{mites}$)**.** *Let, B_H, B_M be the number of healthy and mite-infested adult bee. We capture the dynamics of mite infestation in adult bees as follows:*

$$\dot{H} = c_1 B p(H) - c_2 B e(H) \tag{3}$$

$$\dot{B}_M = c_4 Q p(H) e(H)\, B_M/B \; - f_M(H) B_M - c_{10} u_M B_M \tag{4}$$

$$\dot{B}_H = c_4 Q p(H) e(H)(1 - B_M/B) - f(H) B_H + c_{10} u_M B_M \tag{5}$$

$$B := B_H + B_M \tag{6}$$

The mite-infested and healthy bee dynamics (Eqns. 4–5) follow the same principle as the mite-free model (Eqn. 2), with two additions: mite-control treatment (last terms in Eqns. 4–5) and the mite fraction B_M/B, which captures infestation spread. Mite-infested bees have a higher death rate (c_9) than healthy bees (c_8). We assume the same starvation-induced death rate, c_8', and capture mite-infested bee mortality using the honey-dependent function $f_M(H)$ (similar to $f(H)$, Figure 2a).

3.2.1 Equilibrium Points and Stability Analysis

We present the analytically tractable steady-state solution of DynaBee$_{mites}$ for the linear case. The mite-free equilibrium is the only biologically feasible stable equilibrium.

Lemma 3. *For strictly positive model parameters, $c_i > 0, \forall i$, any biologically feasible equilibrium*

$$(H^*, B_M^*, B_H^*) = (c_1 H_M/(c_1 + c_2), 0, c_4 Q\, p(H^*) e(H^*)/f(H^*))$$

corresponding to a living colony is stable, where,

$$p(H^*) = c_2/(c_1 + c_2), e(H^*) = c_1/(c_1 + c_2) \text{ and } f(H^*) = c_8 c_1/(c_1 + c_2) + c_8' c_2/(c_1 + c_2)$$

Discussion. Note that DynaBee$_{mites}$ reduces to DynaBee$_0$ in case of mite-free equilibrium. For example, any living colony converges to the stable equilibrium $(H^*, B_M^*, B_H^*) = (6.1, 0, 5.6)$ for $c_1 = 0.019$, $c_2 = 0.0122$, $c_4 = 0.329$, $c_8 = 0.01$, $c_8' = 2c_8$, $c_9 = 0.03$ and $c_{10} = 0.042$, ; consistent with sustainable honey reserves and a healthy bee population in a mite-free environment.

3.3 With Larvae & Mites - DynaBee

Next, we propose a more realistic version that includes the missing larval state in both DynaBee$_0$ and DynaBee$_{mites}$.

The transitions mirror known biological processes: e.g., eggs (E) hatch into larvae (L), a fraction of which become mite-infected (L_M) in proportion to infected adult bees (B_M); larvae develop into adults inheriting their infection status ($L_M \to B_M$ and $L_H \to B_H$); infected adult bees may recover if treatment is applied (u_M). Queen presence (Q) and treatment events (u_M) enter as binary inputs. Here, $p(H)$ accounts for the space occupied by honey, eggs, and larvae. We similarly define honey-dependent death functions $g(H)$ and $g_M(H)$ for healthy and mite-infested larvae, assuming the same starvation-induced mortality (Tables 2b, Figure 2a).

3.3.1 Equilibrium Points and Stability Analysis

The steady-state solution of DynaBee for the mite-free and linear case is given below.

Model 3 (DYNABEE). *Our full model is described as follows:*

$$\dot{H} = c_1 B p(H) - (c_2 B + c_3 L) e(H) \tag{7}$$

$$\dot{E} = c_4 Q p(H) e(H) - c_5 E \tag{8}$$

$$\dot{L}_M = c_5 E \frac{B_M}{B} - g_M(H) L_M \tag{9}$$

$$\dot{L}_H = c_5 E \left(1 - \frac{B_M}{B}\right) - g(H) L_H \tag{10}$$

$$\dot{B}_M = c_6 L_M e(H) - f_M(H) B_M - c_{10} u_M B_M \tag{11}$$

$$\dot{B}_H = c_6 L_H e(H) - f(H) B_H + c_{10} u_M B_M \tag{12}$$

$$B := B_H + B_M \tag{13}$$

$$L := L_H + L_M \tag{14}$$

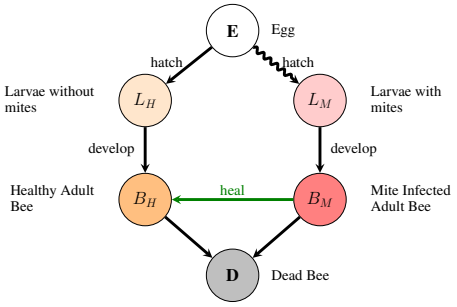


Figure 4: Proposed state transitions of bees for Eqns. 7-12. Green and wavy arrows indicate human intervention and variable transition rate, respectively.

Lemma 4. *For strictly positive model parameters, $c_i > 0, \forall i$, any biologically feasible equilibrium $(H^*, E^*, L_M^*, L_H^*, B_M^*, B_H^*) = (H^*, A^*/c_5, 0, A^*/g(H^*), 0, c_6 A^*/(g(H^*) f(H^*)))$ corresponding to a living colony is stable, where,*

$$p(H^*) = 1 - (H^* + L_H^* + E^*)/H_M, e(H^*) = H^*/H_M, A^* = c_4 Q\, p(H^*) e(H^*)$$

$$g(H^*) = c_7 H^*/H_M + c_7'(1 - H^*/H_M), f(H^*) = c_8 H^*/H_M + c_8'(1 - H^*/H_M)$$

and H^* satisfies, $c_1 p(H^*) - c_2 e(H^*) = c_3 f(H^*)/c_6$

Discussion. As an example, for the parameter values $c_1 = 0.073$, $c_2 = 0.0122$, $c_3 = 2c_2$, $c_4 = 0.561$, $c_5 = 0.33$, $c_6 = 0.048$, $c_7 = 0.0042$, $c_7' = 2c_7$, $c_8 = 0.012$, $c_8' = 2c_8$, $c_9 = 0.057$, $c_9' = 0.133$ and $c_{10} = 0.027$, any living colony converges to the stable mite-free equilibrium $(H^*, E^*, L_M^*, L_H^*, B_M^*, B_H^*) = (3.2, 0.1, 0, 4.8, 0, 3.7)$, consistent with expert expectation.

4 OPTIMIZATION ALGORITHM

In this section, we present our fitting and forecasting algorithm, DYNABEE-Fcast. Our goal is to estimate the parameters for reconstruction and forecasting of hive measurements $\{H, L, B_M, B_H\}$. We take into account the hidden states $\{E, L_M, L_H\}$ and estimate (i) the hive-specific initial conditions (e.g., $H(0), E(0)$, etc.) and (ii) the global parameters (e.g. $c_1, c_2, \ldots$). This optimization function minimizes the reconstruction error and estimates $\{E, L_M, L_H\}$ using our proposed DYNABEE.

4.1 OPTIMIZATION OBJECTIVE

Next, we define our optimization objective for our proposed model, DYNABEE. Our goal is to minimize the reconstruction error for $\{H, L, B_M, B_H\}$ and estimate (i) the hive-specific initial conditions, $X_0 = \{H(0), E(0), L_M(0), L_H(0), B_M(0), B_H(0)\}$ and (ii) the shared parameters, $C = \{c_1, c_2, ...\}$ among different hives of similar setup.

$$\underset{C, X_0}{\text{minimize}}\ E = \sum\nolimits_{b=1}^{B} E_H + E_B + E_L \tag{15}$$

where E_* are the individual squared errors, that is $E_H = \sum_{t=1}^{N} (\hat{H} - H)^2$, $E_B = \sum_{t=1}^{N} [(\hat{B}_H - B_H)^2 + \lambda (\hat{B}_M - B_M)^2]$, and $E_L = \sum_{t=1}^{N} (\hat{L}_H + \hat{L}_M - L)^2$. Here, E_L denotes the squared reconstruction error for total larvae, where $\hat{L}_H$ and $\hat{L}_M$ are hidden states evaluated against the observed $L = L_H + L_M$. For DYNABEE$_0$ and DYNABEE$_{mites}$, $E_L = 0, \lambda = 0$ as mite states are not modeled. Next, we present our algorithm, DYNABEE-Fcast for DYNABEE. It can also be used for DYNABEE$_0$ and DYNABEE$_{mites}$ with minimal modification.

4.2 RECONSTRUCTION & FORECASTING ALGORITHM, DYNABEE-FCAST

How can we minimize E to reconstruct hive measurements $\mathcal{D} = \{\mathcal{H}, \mathcal{L}, \mathcal{B}_\mathcal{M}, \mathcal{B}_\mathcal{H}\}$ and estimate the (i) hive specific initial states $\mathcal{X}_0$ and (ii) the shared global parameters $\mathcal{C}$ to be used later for forecasting future hive measurements, $\hat{\mathcal{D}} = \{\hat{\mathcal{H}}, \hat{\mathcal{L}}, \hat{\mathcal{B}}_M, \hat{\mathcal{B}}_H\}$?

Algorithm 1: DYNABEE-Fcast

Input: (i) Hive dataset, $\mathcal{D} = \{\mathcal{H}, \mathcal{L}, \mathcal{B}_M, \mathcal{B}_H\}$, (ii) Sliding window size, W and (iii) Training data, $\mathcal{D}_T$
Output: Forecast, $\hat{\mathcal{D}} = \{\hat{\mathcal{H}}, \hat{\mathcal{L}}, \hat{\mathcal{B}}_M, \hat{\mathcal{B}}_H\}$

1 $\mathcal{X}_0 \leftarrow \mathcal{X}_{0,init}, \mathcal{C} \leftarrow \mathcal{C}_{init}$ /*Initialization*/
2 /*Parameter estimation from $\mathcal{D}_T$*/
3 **while** *not converged* **do**
4 **for** each hive $b \in \mathcal{B}$, solve $\hat{\mathcal{D}}_T^b$ for $\mathcal{X}_0^b$ and $\mathcal{C}$ using Eqns. 7–12
5 Update $\mathcal{X}_0$ & $\mathcal{C}$ using Non-linear optimizer
6 /*One-step ahead forecasting*/
7 **for** *each hive* $b \in \mathcal{B}$ **do**
8 **for** $k = T, \ldots, N-1$ **do**
9 Define sliding window $k - W$
10 **if** $k = T$ **then** $\mathcal{X}_{0,F}^b \leftarrow \hat{\mathcal{D}}_T^b[-1]$
11 **else**
12 **while** *not converged* **do**
13 Sol. $\hat{\mathcal{D}}_{k-W}^b$ for $\mathcal{X}_0^b$ & $\mathcal{C}$ using Eqn. 7 - 12
14 Update $\mathcal{X}_0^b$ using Non-linear optimizer
15 $\mathcal{X}_{0,F}^b \leftarrow \hat{\mathcal{D}}_{k-W}^b[-1]$
16 $\mathcal{X}_{0,F}^b[H, B_H, B_M] \leftarrow \mathcal{D}_{k-W}^b[-1][H, B_H, B_M]$
17 $\mathcal{X}_{0,F}^b[L_M, L_H] \leftarrow \mathcal{D}_{k-W}^b[-1][L] \frac{\mathcal{X}_{0,F}^b[L_M, L_H]}{\mathcal{X}_{0,F}^b[L_M] + \mathcal{X}_{0,F}^b[L_H]}$
18 Solve $\hat{\mathcal{D}}_{k-W+1}^b$ for $\mathcal{X}_0^b$ & $\mathcal{C}$ using Eqn. 7 - 12
19 $\hat{\mathcal{D}}^b \leftarrow \hat{\mathcal{D}}^b \cup \hat{\mathcal{D}}_{k-W+1}^b$

We implement DYNABEE for curve fitting and reconstruction in Algorithm 1. It takes $\{\mathcal{D}, W, \mathcal{D}_T\}$ as inputs and gives $\hat{\mathcal{D}}$ as output:

- $\mathcal{D} = \{\mathcal{H}, \mathcal{L}, \mathcal{B}_\mathcal{M}, \mathcal{B}_\mathcal{H}\}$ is the recorded inspection data consisting of the number of frames of $\{$honey, larvae, bees with mites, healthy bees$\}$ for B hives. W is the length of the sliding window and $\mathcal{D}_T = \{\mathcal{H}, \mathcal{L}, \mathcal{B}_\mathcal{M}, \mathcal{B}_\mathcal{H}\}_T$ is the fraction of dataset used for parameter estimation (training).
- $\hat{\mathcal{D}} = \{\hat{\mathcal{H}}, \hat{\mathcal{L}}, \hat{\mathcal{B}}_M, \hat{\mathcal{B}}_H\}$ is the one-step ahead forecasting of the inspection data for B hives.

Parameter Estimation (lines 1-5). First, we estimate the hive-specific initial conditions $\mathcal{X}_0 = \{\mathcal{X}_0^{b_1}, \ldots, \mathcal{X}_0^{b_N}\}$ and the global shared parameters $\mathcal{C} = \{c_1, c_2, ...c_{10}\}$ by reconstructing $\mathcal{D}_T$ using Eqns. 7-12. Here, N is the number of hives used for parameter estimation.

One-step-ahead Forecasting (lines 6-19). Next, we use the estimated parameter set $\mathcal{C}$ to perform one-step-ahead forecasting. For the hive-specific initial condition, we use a sliding window of size W to reconstruct each hive's data, denoted by $\mathcal{D}_{k-W}^b$, and then use the final reconstructed state in the window to define the initial condition $\mathcal{X}_{0,F}$ for the next point forecast.

For observed states (H, B_M, and B_H), we use their last values in the sliding window as initial conditions. Since only total larvae L is observed, we scale the reconstructed L_M and L_H proportions by the last observed L. For the unobserved egg state, we use the last reconstructed E. These initial conditions are then used with DYNABEE to forecast the next point, $\hat{\mathcal{D}}_{k-W+1}^b$.

5 EXPERIMENTS

Here we present our experiments, to answer the following research questions:

- RQ1 Accurate: How accurately does our model do one-step-ahead forecasting?
- RQ2 Transferable: Can our model parameters estimated from one colony be reused to forecast the dynamics of other colonies observed under the same conditions, timeline, and genotype?
- RQ3 Explainable: Can our model parameters explain the changes in hive dynamics?
- RQ4 Scalable: Does our model perform linearly in time?

Dataset & Experimental Setup. We used two datasets, one of 6 hives over 14 months (June 2024 - Aug 2025), with interventions and mite treatments; the second with 11 hives over 12 months (Aug

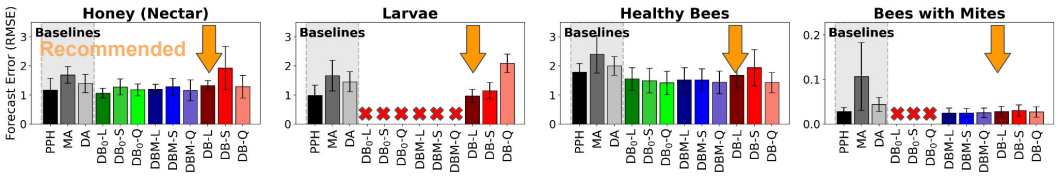


Figure 5: Ablation study: Proposed DYNABEE with linear throttling ties or beats the rest in forecasting accuracy, while it can forecast all the signals of interest. The baselines (in black/gray) have no meaningful parameters. Here, PPH: Prev. Point Heuristic, MA: Moving Average, DA: DeepAR, DB_0: $DYNABEE_0$, DBM: $DYNABEE_{mites}$, DB: DYNABEE, L: Linear, S: Sigmoid, Q: Quadratic.

2024 -Aug 2025) with no interventions and treatments. Combined, these are the **largest and only available** datasets of this kind; it took several technicians over 2 years to collect them. H_M= 10 frames, λ= 100. Colony inspections were typically monthly. We use linear interpolation for missing values (for parameter estimation only). We use Levenberg-Marquardt (LM) to fit our parameters (see Algorithm 1) and introduce biologically motivated constraints for its sensitivity to high-dimensional parameter spaces: $c_3 = 2c_2$, $c_7' = 2c_7$, $c_8' = 2c_8$, and fixed c_2, c_4, and c_5 from the literature.

5.1 RQ1: ACCURATE

Experimental Setup & Baselines. For each hive from the first dataset, we use the first four recorded inspections and eight linearly interpolated points for parameter estimation. The estimated parameters are then used to forecast the next point. The forecasted values are compared against the recorded measurements using RMSE = $\sqrt{(\sum_{i=1}^{n}(x[i] - \hat{x}[i])^2)/n}$, where $x[i]$ and $\hat{x}[i]$ are the recorded and forecasted values of H, L, B_M, or B_H, respectively. We compared DYNABEE against (i) the "previous point" heuristic (AR(1)), (ii) moving average of sliding window size $W = 12$, (iii) DeepAR (hidden size=40, layers=2), (iv) $DYNABEE_0$, (v) $DYNABEE_{mites}$, and (vi) DYNABEE with different throttling functions from Figure 2a.

Results. Figure 1a demonstrates one-step-ahead forecasting (with 95% confidence interval) of (H, L, B_M, B_H) using a sliding window of size $W = 12$ for estimating hive-specific initial conditions. DYNABEE (red lines) forecasts are very close to the recorded points (blue dots). Figure 5 compares the average RMSE of all measurements (H, L, B_M, B_H) among different baselines and our models with different throttling functions. Proposed DYNABEE, with linear throttling, ties or beats the rest in accuracy while forecasting all measurements of interest. $DYNABEE_0$ and $DYNABEE_{mites}$ cannot forecast L and B_M.

5.2 RQ2: TRANSFERABLE

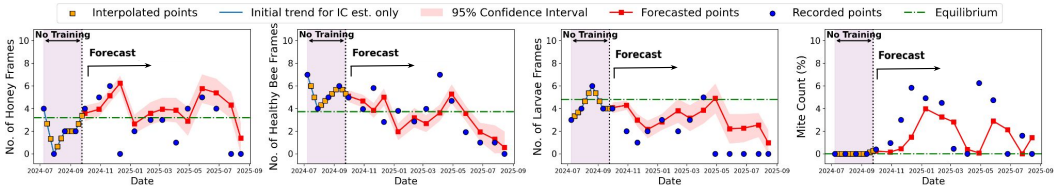


Figure 6: DYNABEE is transferable: the one-day-ahead forecasting for **new** hives (red line) is close to the actual (blue/orange dots). Coloring conventions are the same as in Figure 1a: purple is used for initial conditions (not training); green line for the steady state estimate of DYNABEE.

We use a similar experimental setup as Sec. 5.1 for the second dataset with 11 hives. To evaluate the transferability of the learned colony dynamics, we directly use the shared parameter set estimated from the first dataset and do not re-estimate any model parameters using data from the second dataset. Figure 6 shows the 1-step-away forecasting of DYNABEE. In short, our DYNABEE has similar or better RMSE of all measurements (H, L, B_M, B_H) (Figure 7) among different baselines.

5.3 RQ3: EXPLAINABLE

DYNABEE has the following advantages that black-box methods cannot match:

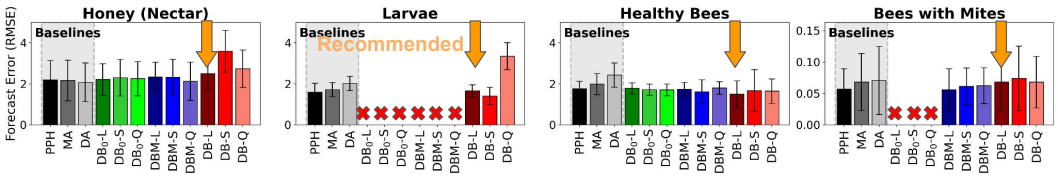


Figure 7: DYNABEE is transferable: Forecasting accuracy for **new** hives, other than the ones DYNABEE was fitted/trained on. DYNABEE with linear throttling ties or beats the rest. Here, PPH: Prev. Point Heuristic, MA: Moving Average, DA: DeepAR, DB_0: $DYNABEE_0$, DBM: $DYNABEE_{mites}$, DB: DYNABEE, L: Linear, S: Sigmoid, Q: Quadratic.

E1) Visualization: We can even plot the 2-d phase-plots for $DYNABEE_0$, as shown in Figure 3. There, we show the two variables, honey H and bees B; the arrows show the direction that the system will move to; the red and blue curves are the so-called null-clines (curves where one of the derivatives is zero); their intersection(s) are the equilibrium points. In Fig 3 we have two such points:

- one is the (0,0) (empty hive, no bees, no honey) which is a saddle point (if we add honey, nothing happens; if we add bees, the system takes off).
- second is the purple point, which corresponds to a hive with $B_{eq}, H_{eq} > 0$. Since the arrows are incoming, this signals stable equilibrium, agreeing with beekeepers' intuition of a functional hive.

E2) Meaningful parameters: DYNABEE only needs a handful of parameters, all of which have intuitive meaning (foraging rate, consumption rate, etc.). Domain experts can inspect the fitted values of our model, and obtain confidence, if the values are within the expected ranges (Table 2b).

E3) Steady-state estimation: Our DYNABEE not only estimates the values at each equilibrium point, but also tells whether the equilibrium is stable, unstable, a saddle point, a limit cycle, etc. The stakeholders can check whether the equilibrium points agree with their intuition.

E4) What-if scenarios: How can we forecast under the scenario that the egg-laying rate of the queen bee (c_4) changes such as (i) 0.56 (almost at peak) (ii) 0.25 (at half-rate) (iii) 0.0 (at the lowest)? Our DYNABEE can easily do that, in contrast to black-box forecasting methods (deep-learning, etc). Figure 1b shows the number of larvae frames (L) for several values of the queen-fertility rate c_4 - as expected, the higher c_4, the more the larvae.

5.4 RQ4: SCALABLE

Depending on the model, our code takes from seconds to minutes on a stock laptop, for a dataset with several months' worth of data. Moreover, our optimization (Sec. 4.1) and fitting algorithm 1 are linear on the number of time-ticks.

6 CONCLUSIONS

Our goal is to develop a white-box method to understand and predict colony behavior, in the presence of mites and mite-treatment. Our main insight is to develop a dynamical system from first principles, in collaboration with entomologists. The resulting DYNABEE has the following properties:

- Accurate: Applied on real data, DYNABEE achieves good accuracy (see Figure 1a).
- Explainable: DYNABEE needs only a handful of parameters with physical interpretation (honey-bee death rate, e.t.c.). Moreover, it can give steady-state estimations and their stability (see Figure 1b), often analytically (see Lemma 4) - a major advantage over black-box baselines.
- Scalable: DYNABEE is scalable, being linear on the input size.
- Transferable: it leads to accurate predictions even for new colonies (see Sec. 5.2).

For brevity, some interesting observations had to be pushed to Appendix A.3.

Reproducibility: Our data and source code is on `https://anonymous.4open.science/r/DynaBee-26EC/`.

REFERENCES

Annika Agatz, Roland Kuhl, Mark Miles, Thorsten Schad, and Thomas G Preuss. An evaluation of the beehave model using honey bee field study data: Insights and recommendations. *Environmental Toxicology and Chemistry*, 38(11):2535–2545, 2019.

Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, et al. Chronos: Learning the language of time series. *arXiv preprint arXiv:2403.07815*, 2024.

Abdul Fatir Ansari, Oleksandr Shchur, Jaris Küken, Andreas Auer, Boran Han, Pedro Mercado, Syama Sundar Rangapuram, Huibin Shen, Lorenzo Stella, Xiyuan Zhang, et al. Chronos-2: From univariate to universal forecasting. *arXiv preprint arXiv:2510.15821*, 2025.

Matthias A Becher, Volker Grimm, Pernille Thorbek, Juliane Horn, Peter J Kennedy, and Juliet L Osborne. Beehave: a systems model of honeybee colony dynamics and foraging to explore multifactorial causes of colony failure. *Journal of applied ecology*, 51(2):470–482, 2014.

Matthew Betti, Josh LeClair, Lindi M Wahl, and Mair Zamir. Bee++: An object-oriented, agent-based simulator for honey bee colonies. *Insects*, 8(1):31, 2017.

George EP Box, Gwilym M Jenkins, Gregory C Reinsel, and Greta M Ljung. *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

Johanne Brunet and Fabiana P Fragoso. What are the main reasons for the worldwide decline in pollinator populations? *CABI Reviews*, January 2024.

Padraig Davidson, Michael Steininger, Florian Lautenschlager, Konstantin Kobs, Anna Krause, and Andreas Hotho. Anomaly detection in beehives using deep recurrent autoencoders, 03 2020.

Elisabeth J Eilers, Claire Kremen, Sarah Smith Greenleaf, Andrea K Garber, and Alexandra-Maria Klein. Contribution of pollinator-mediated crops to nutrients in the human food supply. *PloS one*, 6:1–6, June 2011.

Jürgen Groeneveld, Richard Odemer, and Fabrice Requier. Brood indicators are an early warning signal of honey bee colony loss—a simulation-based study. *Plos one*, 19(5):e0302907, 2024.

James D Hamilton. *Time series analysis*. Princeton university press, 2020.

Honey Bee Health Coalition. Tools for Varroa Management Guide. Technical report, 2022. URL `https://honeybeehealthcoalition.org/resources/varroa-management/`.

Mst Hossain, Christos Faloutsos, Boris Baer, Hyoseung Kim, and Vassilis J Tsotras. EBV: Electronic bee-veterinarian for principled mining and forecasting of honeybee time series. In *SIAM SDM*, 2024.

David S Khoury, Andrew B Barron, and Mary R Myerscough. Modelling food and population dynamics in honey bee colonies. *PloS one*, 8(5):e59084, 2013.

Bryan Lim, Sercan Ö Arık, Nicolas Loeff, and Tomas Pfister. Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International journal of forecasting*, 37(4): 1748–1764, 2021.

Yasuko Matsubara, Yasushi Sakurai, and Christos Faloutsos. Autoplait: automatic mining of co-evolving time sequences. In *SIGMOD Conference*, pp. 193–204. ACM, 2014.

Simon G Potts, Vera Imperatriz-Fonseca, Hien T Ngo, Marcelo A Aizen, Jacobus C Biesmeijer, Thomas D Breeze, Lynn V Dicks, Lucas A Garibaldi, Rosemary Hill, and Josef Settele. Safeguarding pollinators and their values to human well-being. *Nature*, 540(2):220–229, December 2016.

Sara Rodney and John Purdy. Dietary requirements of individual nectar foragers, and colony-level pollen and nectar consumption: a review to support pesticide exposure assessment for honey bees. *Apidologie*, 51(2):163–179, 2020.

Olav Rueppell, Cédric Bachelier, M Kim Fondrk, and Robert E Page Jr. Regulation of life history determines lifespan of worker honey bees (apis mellifera l.). *Experimental gerontology*, 42(10): 1020–1032, 2007.

David Salinas, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. Deepar: Probabilistic forecasting with autoregressive recurrent networks. *International journal of forecasting*, 36(3):1181–1191, 2020.

Isabel Schödl, Richard Odemer, Matthias A Becher, Stefan Berg, Christoph Otten, Volker Grimm, and Jürgen Groeneveld. Simulation of varroa mite control in honey bee colonies without synthetic acaricides: Demonstration of good beekeeping practice for germany in the beehave model. *Ecology and Evolution*, 12(11):e9456, 2022.

Oleksandr Shchur, Ali Caner Turkmen, Nick Erickson, Huibin Shen, Alexander Shirkov, Tony Hu, and Bernie Wang. Autogluon–timeseries: Automl for probabilistic time series forecasting. In *International Conference on Automated Machine Learning*, pp. 9–1. PMLR, 2023.

Nathalie Steinhauer et al. United states honey bee colony losses 2022-2023: Preliminary results from the bee informed partnership. 2024.

Ming Wang, Frederic Tausch, Katharina Schmidt, Matthias Diehl, Silvio Knaebe, Holger Bargen, Lukas Materne, Jürgen Groeneveld, and Volker Grimm. Honeybee pollen but not nectar foraging greatly reduced by neonicotinoids: Insights from ai and simulation. *Computers and Electronics in Agriculture*, 221:108966, 2024.

Chin-Chia Michael Yeh, Yan Zhu, Liudmila Ulanova, Nurjahan Begum, Yifei Ding, Hoang Anh Dau, Diego Furtado Silva, Abdullah Mueen, and Eamonn Keogh. Matrix profile i: All pairs similarity joins for time series: A unifying view that includes motifs, discords and shapelets. In *ICDM*, pp. 1317–1322, 2016.

Byoung-Kee Yi, Hosagrahar V Jagadish, and Christos Faloutsos. Efficient retrieval of similar time sequences under time warping. In *ICDE*, pp. 201–208, 1998.

A APPENDIX

A.1 BACKGROUND - DYNAMICAL SYSTEMS

Since one of our main contributions is the development of a dynamical system to model the honeybee behavior, here we review the main concepts from dynamical systems: definition, equilibria, stability, and phase plots. All these concepts are vital for the explainability of our DYNABEE.

Time-invariant Dynamical System: This is a set of (first order) differential equations, that do *not* depend on time. For example, Eq 1-2, repeated here in simplified form (without the throttling factors that we describe in the next section), define such a system.

$$\partial H/\partial t \equiv \dot{H} = \overbrace{c_1\, B}^{foraging} - \overbrace{c_2\, B\, H}^{consumption} \tag{16}$$

$$\partial B/\partial t \equiv \dot{B} = \underbrace{c_4}_{birth} - \underbrace{c_7 B}_{death} \tag{17}$$

From now on, we will use the 'dot' notation for the derivative, since it is more compact. The symbols are explained in Table 2b: the amount of honey ($H = H(t)$) grows with the number of bees $B = B(t)$ (with rate $c_1 - c_2 H$ - foraging minus consumption rate - notice that consumption decreases if there is little available honey); bees are born with rate c_4 and die with rate c_7.

In general, consider n variables $\vec{x} = (x_1, ...x_n)$ that change over time. For clarity, we dropped the t argument: $x_1 = x_1(t)$ etc. Then, a time-invariant (a.k.a. autonomous) dynamical system obeys the definition:

Definition 4 (Time-invariant dynamical system)**.** *Such a system consist of n ordinary differential equations, as below:*

$$\dot{x}_i = F_i(x_1, ...x_n) \quad i = 1, 2, \ldots, n$$

with initial condition $\vec{x}(0) = (x_1(0), ...x_n(0))$

Equilibrium points: These points are the steady-state solutions of the system. In the above example, the equilibrium points are the solutions $\vec{x}_{eq}$ to the equations

$$F_i(\vec{x}) = 0 \quad i = 1, 2, \ldots, n$$

Example: For our simple example (Eq. 16-17) the equilibrium equations are obtained by setting $\dot{H} = 0$ and $\dot{B} = 0$:

$$c_1 B_{eq} - c_2 B_{eq} H_{eq} = 0 \tag{18}$$

$$c_4 - c_7 B_{eq} = 0 \tag{19}$$

eventually leading to equilibrium values $H_{eq} = c_1/c_2$ and $B_{eq} = c_4/c_7$.

Stability: The stability of an equilibrium point is completely determined by the eigenvalues of the Jacobian matrix. The i, j entry of the Jacobian is given by: $Jacobian[i, j] = \partial F_i/\partial x_j$ at the equilibrium point $\vec{x_{eq}}$. See sec. 3.3 for the Jacobians of our proposed dynamical system. An equilibrium point $\vec{x_{eq}}$ is *stable* if all eigenvalues have negative real parts; unstable if the real parts are positive; saddle point if some are positive and some are negative; limit cycle if they are complex.

The reason we mention equilibria points is that they provide *sanity checks* for our models: We expect that the equilibria points of our model to be stable and that their arithmetic values to agree with the intuition of the domain experts.

Phase plot. Figure 3(a) shows a phase plot: for a 2-variable dynamical system, like the upcoming Eq 1-2, we plot H versus B, and for several grid points, we also plot the derivatives $\dot{H}, \dot{B}$. Such a plot is a great debugging and explainability tool: the arrows should be pointing towards the nearest stable equilibrium point. The so-called *null-clines* are the curves where the derivatives are zero (see blue and red lines in Fig. 3).

A.2 Proof of Lemma 1

Proof. In Eqn. 1, set $\dot{H} = 0$,

$$B[c_1 p(H) - c_2 e(H)] = 0 \tag{20}$$

For any living colony, the equilibrium for bees is positive, $B^* > 0$, so, from Eqn. equation 20,

$$c_1(1 - \frac{H^*}{H_M}) = c_2 \frac{H^*}{H_M} \Rightarrow \boxed{H^* = \frac{c_1}{c_1 + c_2} H_M}$$

Now, in Eqn. 2, set $\dot{B} = 0$,

$$c_4 Q\, p(H)\, e(H) - B\, f(H) = 0 \Rightarrow \boxed{B^* = \frac{c_4 Q\, p(H^*)\, e(H^*)}{f(H^*)}}$$

The Jacobian is,

$$J(H^*, B^*) = \begin{bmatrix} -B^* \frac{c_1+c_2}{H_M} & 0 \\ * & -f(H^*) \end{bmatrix}$$

The eigenvalues are, $\lambda_1 = -B^* \frac{c_1+c_2}{H_M}, \lambda_2 = -f(H^*)$

All parameters are positive, so $\lambda_1 < 0, \lambda_2 < 0$, means this equilibrium is stable. ■

A.3 Discussion

Here we list some of the highlights of DynaBee, as well as some lessons learned.

Observation 1 (Asymptotic Behavior)**.** *Thanks to our dynamical-system approach, we know what is the steady-state behavior of our equations.*

Discussion: Not only DynaBee can estimate the equilibria, but it can also tell whether they will be stable, unstable, saddle-points or some other form. A domain expert can easily check whether the implied equilibria make biological sense, and thus we obtain confidence on the fitted model.

Observation 2 (White-box pays off)**.** DynaBee *is a white-box method, with only a few parameters to be determined from the data.*

Discussion: The few parameters of DynaBee all have physical meaning (foraging rate, honey-consuption rate, etc), which adds to the explainability and sanity-checks of the method: the estimated parameters that we have derived earlier, all agree with the intuition of domain experts. This is in stark contrast with the majority of forecasting and modeling methods (with or without deep learning), where parameters have no physical meaning.

Observation 3 (Simpler may be better)**.** *Linear throttling seems to be as accurate or better than the more elaborate throttling versions (see Figure 5).*

Discussion: This is nice, since the linear throttling is easier to solve analytically. Moreover, it is often the case that simpler models indeed are as accurate as more complicated ones, which may suffer from overfitting.

Observation 4 (Transferability)**.** *As shown earlier Figures 7, 6 the parameter values that* DynaBee *fitted in Dataset 1 were able to do good forecasting for Dataset 2, without the need to re-fit/re-train.*

Discussion: This is a valuable observation, confirming our intuition that our parameters will hold for several settings, since it is the same species (*Apis Mellifera*), and the parameters depend on the biology (foraging ability, honey consumption rate, mite infection rate, etc). Again, this is one more reason to use a white-box approach, where we know when/whether the fitted model of DynaBee will work.