2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

# Toward Perpetual Occlusion-Aware Observation of Comb States in Living Honeybee Colonies

Jan Blaha[1], Tomáš Vintr[2], Jan Mikula[3,1], Jiří Janota[1], Tomáš Rouček[1], Jiří Ulrich[1], Fatemeh Rekabi-Bana[2], Laurenz Alexander Fedotoff[4], Martin Stefanec[4], Thomas Schmickl[4], Farshad Arvin[2], Miroslav Kulich[3], and Tomáš Krajník[1]

***Abstract*— Honeybees are one of the most important pollinators in the ecosystem. Unfortunately, the dynamics of living honeybee colonies are not well understood due to their complexity and difficulty of observation. In our project "RoboRoyale", we build and operate a robot to be a part of a bio-hybrid system, which currently observes the honeybee queen in the colony and physically tracks it with a camera. Apart from tracking and observing the queen, the system needs to monitor the state of the honeybee comb which is most of the time occluded by workerbees. This introduces a necessary tradeoff between tracking the queen and visiting the rest of the hive to create a daily map. We aim to collect the necessary data more effectively. We evaluate several mapping methods that consider the previous observations and forecasted densities of bees occluding the view. To predict the presence of bees, we use previously established maps of dynamics developed for autonomy in human-populated environments. Using data from the last observational season, we show significant improvement of the informed comb mapping methods over our current system. This will allow us to use our resources more effectively in the upcoming season.**

## I. INTRODUCTION

The advances in additive manufacturing and miniaturization enabled fast prototyping of robots tailored to be integrated into natural systems [1]. This opened domains beyond traditional agri-tech applications, resulting in robots interacting with animals in their natural ecosystems [2]. These robots can carry out not only ecological monitoring [3]–[5], but also actively support nature conservation through maintenance [6] and recovery of declining ecosystems [7]. Active support of the ecosystems is going to gain importance, especially due to the ongoing ecological crisis [8], [9].

We investigate the use of robots to support honeybee colonies, which are considered keystone species crucial for ecosystem stability. In the EU project RoboRoyale, we aim to use robotic surrogates to interact with honeybee queens to improve the efficiency of honeybee colonies' pollination. To assess the effect of the interactions, the system has to monitor not only the queen but also the worker bees, the brood, and the comb cells. To enable these observations, we developed a gantry robotic system with a moving camera [10]. It is capable of continuously tracking the queen [11] and capturing images of the worker bees and the comb cell contents with unprecedented detail and extent of observations. However, our robot cannot observe the entire colony state at once. This requires to consider the dynamics of the queen, worker bees, and the comb when planning and executing the observations.

Since the full brood evolution takes about three weeks and

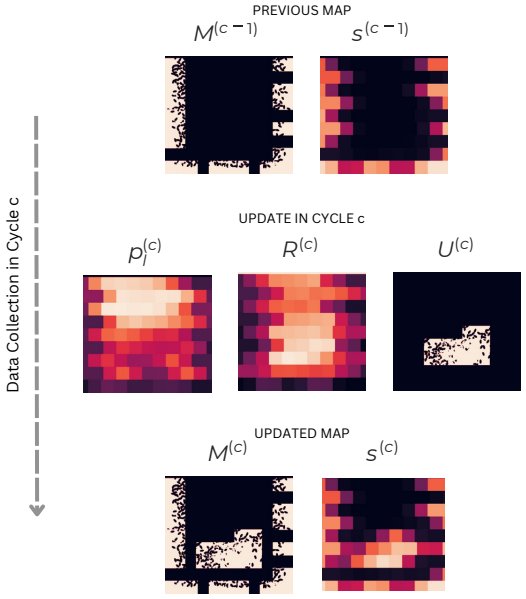


Fig. 1: Example visualization of the second cycle ($c = 2$) of predictive mapping with $a = 0.1$. Starting with a map from the previous cycle $\mathbf{M^{(c-1)}}$ (white color repr. observed parts of comb), and respective states of coverage $\mathbf{s^{(c-1)}}$ (lighter color indicates higher coverage), the system decides which locations to visit in cycle $c$. Predictive methods give for time $t$ of the cycle predictions of the density of bees $\mathbf{p_t^{(c)}}$ (darker colors represent more empty parts of the comb). From $\mathbf{s^{(c-1)}}$ and $\mathbf{p_t^{(c)}}$, the system generates a ranking of locations $\mathbf{R^{(c)}}$ (lighter colours represent lower ranks) and then chooses $a\%$ of scans to visit, collects data $\mathbf{U^{(c)}}$ and combines it with $\mathbf{M^{(c-1)}}$ into $\mathbf{M^{(c)}}$ which gives new states of coverage $\mathbf{s^{(c)}}$.

[1] Faculty of Electrical Engineering, Czech Technical University in Prague, Czechia name.surname@fel.cvut.cz
[2] Durham University, Computer Science Department, United Kingdom name.surname@durham.ac.uk
[3] Czech Institute of Informatics, Robotics and Cybernetics, Czech Technical University in Prague, Czechia name.surname@cvut.cz
[4] Artificial Life Lab, Department of Zoology, Institute of Biology, University of Graz; Graz, Austria name.surname@uni-graz.at

The code for the experiments in this paper is available at https://gitlab.roboroyale.eu/paper-experiments/2024_iros_toward_observation.

filling the combs with honey takes several days, observations of the comb cells once per day are sufficient to capture their dynamics. The queen's activity breaks down into intervals of patrolling, when she walks around the comb, inspects the cells and lays eggs, and resting, where she stays at one place for several minutes. During resting, the queen either actively interacts with the worker bees who feed and groom her, or she remains motionless. These resting periods can be detected by the vision-based tracking, and the robot has a few minutes to turn its attention away from the queen. However, a survey of the entire comb requires a combination of 80 to 90 images captured at different locations of the comb. Thus, the resting periods are not long enough to survey the entire comb without risking missing important interactions before the queen resumes her movement. This is further aggravated by the worker-bees occluding substantial parts of the combs, which requires either staying at each scanning location long enough for the bees to move around or revisiting these locations several times throughout one day.

However, the activity of the bees is partially governed by the proximity of the queen, the distribution of the brood, and the influx of nutrients, which is subject to the diurnal cycle. Thus, the density of the worker bees at different areas of the hive changes could be potentially modeled and forecasted. This allows us to predict which observations are most likely to contain cells that have not been observed during the given day.

In this paper, we investigate methods that allow us to plan the observations to survey the comb in an efficient manner. We aim to reduce the number of observations and consider the time spent performing them while achieving observations of the contents of most of the cells in the hive each day.

Using a dataset from a living beehive colony covering several weeks, we assessed ten different methods. Two of these are uninformed, and eight are informed, of which one is greedy and the rest predictive, using maps of spatio-temporal dynamics adopted from research on long-term autonomy, where they were developed to model people presence for robot navigation [12], [13]. On this problem we demonstrate the value of such techniques also for description of other then human behaviors.

## II. RELATED WORK

There were efforts to capture data about living bee colonies using modern technologies and machine learning algorithms, like the trajectories of worker-bees using markers [14] or later without them [15]. While these works present significant steps toward understanding honeybee behavior on the scale of the whole colony, they do not yet capture long observations and show data in horizons under $1\,\mathrm{h}$. The work of [11] opened the effort for truly long-term observations inside the hive, with marker-based visual tracking of the queen. In the last version our system is tracking the queen and various KPIs of the colony using a mounted movable camera allowing for detailed observation [10].

Collecting data under possible occlusions poses two key challenges for robots: *safe navigation* and *task efficiency*. The *navigation* challenge may involve maneuvering through unknown environments with occlusions, such as when grasping partially visible objects [16] or ensuring safety by anticipating encounters with pedestrians [17] or cars [18]. Solutions typically rely on the robot's ability to minimize occlusions through tailored movements and additional data collection or on robust planning considering potential information gaps.

The second challenge, *task efficiency*, can be approached through two paradigms: *inspection* or *search*. In inspection tasks, the objective is to minimize resources (e.g., time or energy) required to collect all data or maximize the reward associated with collecting partial data while on a resource budget. In the complete-graph domain, the former scenario leads to the well-known *traveling salesman problem* (TSP), while the latter leads to the *orienteering problem* (OP) [19].

Under the *search paradigm*, collecting more data early throughout the task is most important, optimizing for the scenario when resources are highly likely to be limited, but the exact budget value is not known in advance. On graphs, this is known as the *graph search problem* (GSP) [20].

Both paradigms may apply to our case. Inspection is the more natural one, as it combines completing the data-collection task while minimizing resources. Particularly, in this paper, we optimize for the minimum total number of collection cycles. In addition, we use the traversed distance (which is a proxy for completion time) as an auxiliary statistic. In the future, we consider an extension to the OP formulation. The search paradigm becomes applicable when the data-collection task may be interrupted due to the end of the queen's resting period, an event not known in advance. For this scenario, we present preliminary results as a stepping stone for future work. The search paradigm has not been considered in our main stream of experiments due to unfinished integration with the rest of the system.

Maps of Dynamics (MoDs) [21] are robotic maps that enable the prediction of an environmental state outside of the robot's perceptual range and forecast into the future. There exists a large scale of different approaches, which can be categorized by the types of dynamics they model, information that can be extracted from observations, a form of representation, and areas of applications [12].

In this work, we rely on spatio-temporal MoDs [22] that model active, non-directional dynamics with independent area-based observations representing the dynamics by a temporal model IV-A suitable for task planning [13]. The original idea that constituted this branch of MoDs states that robotic maps need to adapt to the environment continuously over time [23]. Over time, it evolved from a prompt reaction to the structural changes [23] to a forecasting of changes occurring repeatedly [24]. This approach was then broadened to modeling of dynamics caused by human actions [25] and later human presence [26], people density [27], and human flows [13] in urban, human-populated environments.

An integration of spatio-temporal MoD to enhance a decision-making process in a bio-hybrid system is, therefore, a pioneering work. In the following experiments, the methods are confronted with spatial and temporal structures developed

by insects, which can differ substantially from principles of human behavior for which the methods were developed.

## III. PROBLEM

Our hive is adapted for the observation of bee colonies. It is made of two $420\,\mathrm{mm} \times 220\,\mathrm{mm}$ wooden combs on top of each other with a glass panel on both sides. The hive is located indoors, but it is connected to the exterior with a plastic tube so the bees can fly outside.

### A. Observation System

Rather than observing the entire honeybee colony in low detail as in [11], our system is designed to provide a high level of detail over a small area anywhere in the hive. This is achieved by a positionable camera of $1920\,\mathrm{px} \times 1080\,\mathrm{px}$ resolution and $30\,\mathrm{Hz}$ capture rate. For simplicity, we treat each side of the hive separately and label them "side 0" and "side 1".

### B. Incremental Comb Observations

Our goal is to collect a complete image of the underlying comb without occlusions on a daily basis. In the current setup, the system does so by recording a complete scan of the whole hive at every opportunity given by the queen's resting. This means going to every position defined on a regular grid covering the comb and collecting local data. The collected data is combined with previous observations to filter bee occlusions. We consider the level of coverage to be the observed comb area for each position individually. The task then is to optimize the selection of locations to gain high coverage with a limited number of observations. An example of how the whole cycle looks for a predictive method is shown in Fig. 1.

The regular rectangular grid of scanning consists of tiles $l_i$ for $i = 1, \ldots, N_l$ given by the position of the camera and defined by the area of the comb it can see given its field of view. The tiles are not exclusive and can overlap, which introduces dependencies in data collection between locations. Each observed scan $S$ at time $t$ is a mapping $S_t(i) = T_{t,i}$, where $T_{t,i} = \left((x, y, \phi)_{t,i,j}\right)_{j=0}^{N_i}$ is a set of bee observations at the tile $i$.

Each day $d$, a certain number of scanning cycles $c = 1, \ldots, S_d$ is conducted. Over this time, in the form of a matrix, a composed rasterized map $M$ of the comb under the bee occlusions is cumulatively formed. We denote $M^{(c)}$ the map after cycle $c$. The state vector $\mathbf{s}^{(c)}$, $s_l^{(c)} = |A_{l,\text{covered}}^{(c)}|/|A_{l,\text{all}}^{(c)}|$, where $A_l$ is a block of $M$ representing the cameras field-of-view at location $l$, stores the proportion of area $A$ captured free of bees at each location after cycle $c$.

Each mapping method then for a cycle $c$ proposes a ranking of locations to visit $R^{(c)}$, formally a permutation of $\{1, \ldots, N_l\}$, given $\mathbf{s}^{(c)}$ and time $t$ at the start of the cycle. According to this ranking first $a \cdot 100\%$ of locations $L^{(c)} = l_{R^{(c)}(1)}, \ldots, l_{R^{(c)}(aN_l)}$ are visited; we further refer to $a$ as "areas ratio". An empty copy of $M$ denoted $U^{(c)}$, is formed by overlaying over $U$ ellipses representing detected bees at $(x, y)$ with orientation $\phi$ for each detection in $T_{t,l}$ and at location $l \in L$. This way, $U^{(c)}$ is undefined outside of visited locations, and on them represents occlusions by bees. The $U^{(c)}$ is combined with $M^{(c-1)}$ to form $M^{(c)}$ so, that $\forall l : A_l^{(c)} = A_l^{(c-1)} \vee \neg B_l^{(c)}$, where $A_l$ is a block of $M$ and $B_l$ block of $U$ representing the cameras field-of-view at location $l$. Using this rule, $M$ cumulatively stores the areas seen without occlusions.

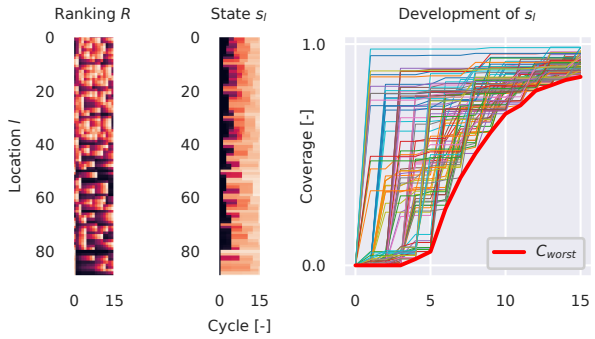


Fig. 2: Example of evolution of the state vector $\mathbf{s}$ and the corresponding $C_{\text{worst}}$ statistic. The Area Ratio $a$ was set to 0.1. The left graph represents the evolution of ranking in every one of the 90 locations (rows) during every cycle (columns). The lighter the color, the lower the rank. The middle graph represents the evolution of states, again, rows representing locations and columns cycles. The lighter the color, the higher the coverage. The right graph then represents an evolution of coverage of every location after every cycle (thin lines). The thick red line highlights an evolution of a minimum coverage over all locations $C_{\text{worst}}$.

The individual methods are then fully defined by the way ranking $R^{(c)}$ is derived at each cycle. The methods we consider in this work are fully described in Section IV-B.

## IV. METHODS

### A. Applied Maps of Dynamics

The spatio-temporal MoDs we use are learning algorithms taking in general data in the form of $\{(t, \mathbf{o})_i\}_i^N$, where $t$ is a timestamp, and $\mathbf{o}$ is a vector of observation's characteristics. Upon learning, these algorithms provide a function $f(t)$ forecasting the state of the environment at a chosen time $t$.

Our dataset $\{(t, x, y)_i\}_i^N$ consists of $N$ detections of bees characterized by their position in time, $\mathbf{o} = (t, x, y)$. We consider seven successful MoDs from [13], namely: HistDayGrid, HyT_X_GMM, HyTted_GMM, MeanGrid, WHyTe, WHyTened_kMeans, and time_window_GMM. The names are usually composed of prefixes and postfixes, following the convention in the original paper. Prefix defines a forecasting method while postfix defines the spatial structure of the map—Grid means that the forecasting is carried out in each cell of the predefined grid, kMeans, and GMM means that the grid is derived from the data by applying the corresponding clustering method, and absence of the postfix (WHyTe) stands for a method that models space and time together.

### B. Considered Methods of Locations Ranking

Here, we specify the considered mapping methods, giving the rules for the ranking of locations $R$ to select the best ones to visit. Our mapping methods fall into three main categories:

*1) Uninformed:* We consider two uninformed methods; one takes

$$R_{\text{random}} \sim \text{Permutation}(\{1, \ldots, N_l\}), \tag{1}$$

to be a random reshuffle of location indexes. The second performs a preempted snake-like motion, i.e.

$$R^{(c)}_{\text{sequential}}(x) = (x + c \cdot a \cdot N_x) \mod N_x. \tag{2}$$

*2) Greedy:* Greedy mapping prefers to visit areas that are least covered, so for $R_{\text{greedy}}$ it holds that

$$\forall (i,j) : s_i^{(c-1)} < s_j^{(c-1)} \Rightarrow R^{(c)}_{\text{greedy}}(i) < R^{(c)}_{\text{greedy}}(j). \tag{3}$$

*3) Predictive:* Each of the predictive mapping methods employs a specific MoD model $f_{\text{model}}$ with predictions computed as

$$\forall l : p_l = \begin{cases} \int_{A_l} f_{\text{model}}(t,x,y)dxdy & \text{space-continuous} \\ f_{\text{model}}(t,l) & \text{space-discrete} \end{cases}. \tag{4}$$

From the normalized predictions $p'_l = \frac{p_l}{\sum_l p_l}$, we compute an optimistic estimate of information gain at the location $l$ and time $t$

$$e^{(c-1)}_{\text{model},l} = (1 - s_l^{(c-1)})(1 - p'_l), \tag{5}$$

which is the product of missing coverage and relative expected free space. From the expected information gain, the ranking follows

$$\forall (i,j) : e_i^{(c)} < e_j^{(c)} \Rightarrow R^{(c)}_{\text{model}}(i) < R^{(c)}_{\text{model}}(j). \tag{6}$$

## V. EXPERIMENTS

For our experiments, we use a dataset collected by the system to simulate the selective data collection. The main objective is to capture a daily snapshot of the entire comb area, given shorter time windows and selective data collection. We looked at two basic criteria: the time until certain coverage $s_l$ is reached over all locations $l$ and, complementarily, the progress of the coverage in time. The notion of time has three main interpretations in our work. The first is the number of data collecting cycles, the second is the traversed meters, and finally, the actual time, which depends on the physical parameters of the setup, like the time it takes to collect an appropriate sample. The cycles and meters are motivated by two main modes of data collection—observing and scanning—with cycles combined with the ratio of visited location $a$, which also corresponds to the amount of raw data that is collected and needs to be stored. The observing mode requires the camera to stay for a longer time at each location, e.g., to capture the dynamics of worker-bees and their interactions. Scanning takes one photo at each location, which requires only a short time to collect. Both modes are interesting for our project, so we analyze time in terms of both cycles and traversed meters and then look at the real-time, substituting the parameters of our setup.

For our testing, we kept both sides of our hive separately, as the hardware was set up a bit differently, which allows for control over the camera setup (that is why we have different number of locations for both sides). The daily snapshot was composed separately for each day and varying areas ratio of allowed locations $a$, taken as ten quantiles from $5\,\%$ to $100\,\%$. We analyze the time needed to reach a certain predefined coverage and compare the overall covering performance of chosen methods.

### A. Collected Dataset

A one-month-long dataset of full scans collected by our robotic system [10], containing 729 scans on side 0 and 1108 scans on side 1, was processed. The scans consist of honeybee comb images organized in a regular rectangular grid, taken at 90 locations on side 0 and 80 on side 1. Bees are detected using a fine-tuned YOLOv8 model [28], and their orientation is estimated using a custom convolutional neural network (CNN) with fully connected layers. Both models were presented in [10]. This way, we obtain almost two million bee detections on side 0 and three million detections on side 1. To get training data for the maps of the dynamics, it is necessary to have enough temporal extent of the training data, so we split the dataset into the first 21 days for training of MoDs and the following 8 days for testing covering methods.

### B. Coverage

We define the achieved coverage as a minimum over all locations,

$$C_{\text{worst},d}(i) = \min_l s_l^{(i)}, \tag{7}$$

where $i = 1, \ldots, S_d$ is an index of the scanning cycles performed on day $d$. An example of the evolution of the state vector $\mathbf{s}$ based on ranking $R$ and the resulting $C_{\text{worst}}$ is shown in Fig. 2.

### C. Covering Metrics

Once given the set of selected locations $L^{(c)} = \{l_i\}_{i=1}^{aN}$, we also estimate the cost of the planned data collection. In the current setup, the camera always starts in the bottom left corner of the hive, but for selective collection, the camera would start at the position of the resting queen. Since that is often repetitive, we chose to simulate the starting position by a fixed value of the origin $O = (0.11, 0.312)$, which corresponds to the location where the queen rested the most in our data. We compute the optimal order of visits for $L' = \{O\} \cup L$ as a solution to the TSP problem

$$\pi^* = \arg\min_\pi \sum_i \text{dist}(l_{\pi(i)}, l_{\pi(i+1 \bmod |L'|)}) \tag{8}$$

where $\pi : \{0, \ldots, |L'|\} \to \{0, \ldots, |L'|\}$ is a permutation on the selected locations to minimize the distance traveled to visit all locations and come back to the origin $O$. To solve this problem, we used open-source software PyConcorde [29], wrapping the Concorde TSP solver [30]. Taking the $\pi^{*(c)}$, the corresponding optimal value $o^{(c)}$ and the length of the longer edge from $O$, $lo^{(c)}$, we compute

$o'^{(c)} = o^{(c)} - lo^{(c)}$ which is the length of the optimal open path without return to the origin. We define

$$m^{(c)} = \sum_{i=0}^{c} o'^{(i)}, \tag{9}$$

as the cumulated cost over the individual scanning cycle. In other words, visiting all its selected locations up to cycle $c$, the camera would have to travel $m^{(c)}$ meters.

For our experiments, we chose the threshold on coverage $Th = .99$, which we consider good enough for the daily snapshot. This gives us two criteria:

$$CN = \text{avg}_{d, i > S_d} \min\{\, i \mid C_d(i) > Th \,\}, \tag{10}$$

$$MN = \text{avg}_{d, CN_d > S_d}\, m_d^{(CN_d)}, \tag{11}$$

where $CN$ is the minimal number of cycles (visits), and $MN$ is the number of meters needed to reach the coverage of $Th$ averaged over days in which the method was able to reach $Th$ given that each day there is a different number of available windows for scanning.

Further, we compute the AUC for each combination of areas ratio and day $(a, d)$ defined as

$$AUC_{a,d} = \int_i C_{\text{worst},(a,d)}(i)\, di. \tag{12}$$

To compare the overall performance of our methods, we compare the $AUC$s they produce. We cannot compare the values directly because they are strongly affected by the parameter $a$—the more locations to visit in each cycle, the faster the coverage—so we resort to non-parametric methods based on ranking. As we have several methods and many $(a, d)$ pairs, we use the Quade statistical test for complete block design studies [31], implementation of [32], correcting the p-values for the problem of multiple comparisons with Šidák's correction [33]. We set the significance level to $5\,\%$.

### D. Preliminary Experiments on the Search Paradigm

In the experiments described above, we assess data collection efficiency in terms of the number of collection cycles, which corresponds to the inspection paradigm (recall Sec. II). However, the search paradigm would be more appropriate if we were to consider that the length of data collection time is a random variable given by the duration of the queen's resting. Under this paradigm, collecting more data early is more important than collecting all data in the shortest time, which could be advantageous when the queen's resting time permits only partial completion of the task.

To assess the prospect of the search paradigm, we consider an additional method of selecting the locations to visit based on the GSP solution, obtained using the *multi-start generalized variable neighborhood search* (Ms-GVNS) metaheuristic [34]. In the GSP [20], the locations are represented as vertices in a complete graph, and each vertex $i$ is weighted according to the associated information gain $e_i$. The GSP aims to find:

$$\pi^*_{\text{GSP}} = \arg\min_{\pi} \sum_{i=1}^{|L'|} e_i \sum_{k=1}^{i} \text{dist}(l_{\pi(i-1)}, l_{\pi(i)}). \tag{13}$$

The GSP-based method is evaluated in two sets of *preliminary* experiments where we do not consider the effect of the searching paradigm on the comb mapping but rather how it would compare in the generated trajectories. These experiments aim to show a clear distinction between the inspection and search paradigms using quantitative metrics that could indicate in which situations to apply each paradigm and the potential for improvement obtained by making that choice. For this purpose, we utilize the sets of locations selected by the WHyTened_kMeans method over the course of experiments from Sec. V-B, as we expected this method to perform well based on our prior experience.

The first experiment tested the suboptimality of $\pi^*_{\text{TSP}}$ in terms of information collectible in the time (traveled distance) needed to visit all locations in a potentially very scattered selection $L$. In the second experiment, we look at the potential gain of information along the path if optimizing for the possibility of interruption. We ran both experiments over the sets of locations selected by the WHyTened_kMeans method over the experiments from Section V-B as we expected it to perform well based on our previous work. The first experiment was run for various $a$, the second only for $a = 1$.

We set $cost_{\text{GSP}}(j)$ to be the distance cumulated over steps $j$ and $E_{\text{GSP}}(d) = \sum_{i, cost(i) < d} e_i$ to be the cumulated information along distance traveled by $\pi^*_{\text{GSP}}$. To assess the possible information to be gained by $\pi^*_{\text{GSP}}$ at the same cost $o'$ of $\pi^*_{\text{TSP}}$ we compute the total information collected until budget depletion as $TE_{\text{GSP}} = E_{\text{GSP}}(o')$, and we report the information gap as $IG = (TE_{\text{GSP}} - TE_{\text{TSP}})/TE_{\text{TSP}}$.

To assess the gain in case of interruptions, we compute the information coverage as $IC_*(r) = E_*(r \cdot o')/E_*(cost_*(N_l))$, which represents the total information collected after traveling portion $r$ of TSP solution cost $o'$.

## VI. RESULTS

Here, we show the results of our experiments. First, we look at the overall coverage performance, then at the time-based evaluation in terms of the necessary scans and traversed distance. Finally, we show the results on the possible improvement of the distance cost of the collecting trajectories. Note that presented Fig. 4 and 5 show a gap at $a = .05$ because none of the methods reached the required $Th = .99$; the Random method had trouble even with more locations allowed.

### A. Coverage Comparison

Looking at the performance of the methods across all cycles of all days and for all values of $a$, we compare the values of $AUC$ generated for all pairs $(a, d)$. The results of testing the rankings of the methods according to the $AUC$ criteria are given in Fig. 3. We show where the Quade test, including Šidák's correction, confirmed statistically significant differences.

We see the uninformed methods performing worse than the informed ones. Apart from WHyTened_kMeans, all methods outperform the Greedy approach. At side 0, the

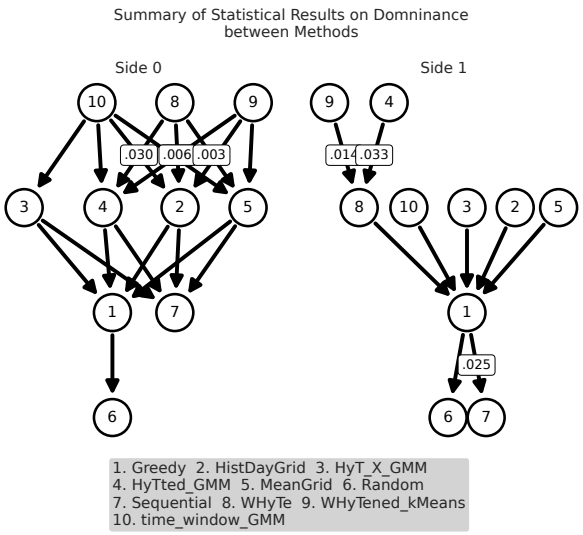


Fig. 3: Diagram showing the DAG of statistically significant differences in $AUC$ of individual methods over all values of $a$ and days $d$. An edge is between two methods if the p-value corresponding to the test of their difference is less than the set significance level of $5\,\%$, and the edge cannot be removed by transitive reduction. For clarity, we explicitly show only those p-values where the $p \geq 1 \times 10^{-3}$. The list of methods is alphabetical, but the edge orientation corresponds to the ordering in the average rank of the methods.

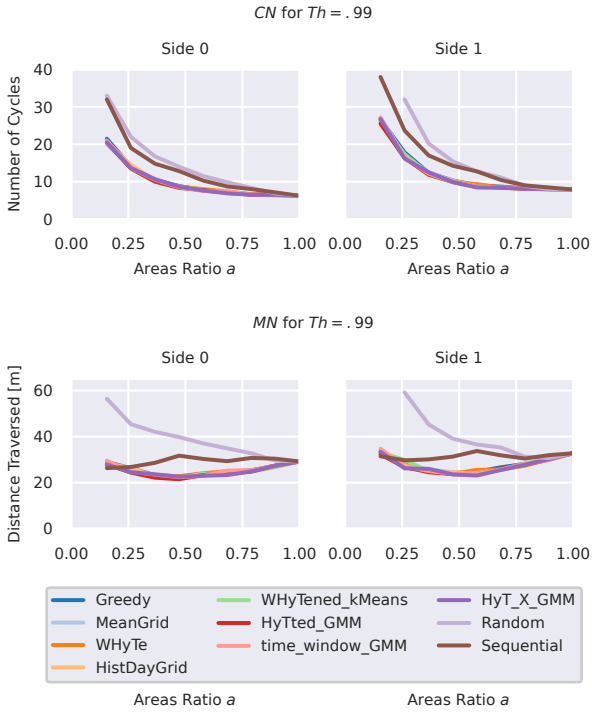


Fig. 4: The number of collection cycles $CN$ (top) and the number of traversed meters $MN$ (bottom) required for each method to reach the coverage of $C = .99$ on each side visiting only $a$ of all locations in every cycle.

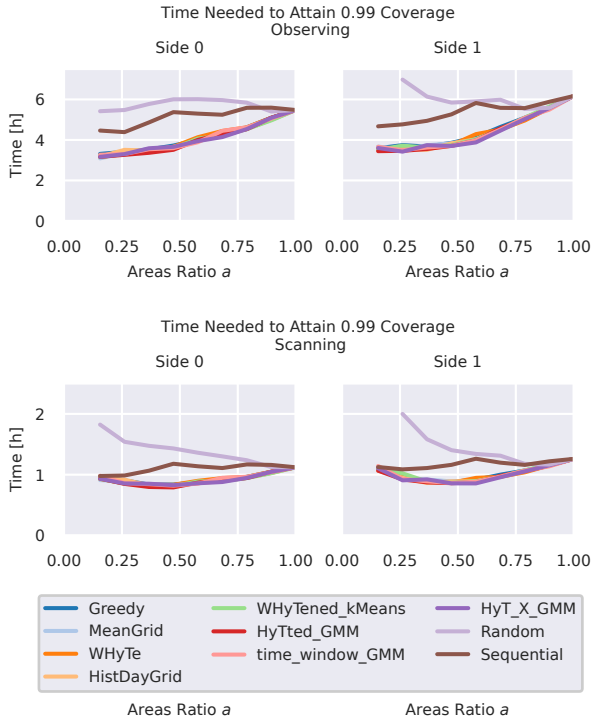


Fig. 5: The amount of time the system would require to reach the coverage of $C = .99$, given that for observation mode, the data collection takes $30\,\mathrm{s}$, for scanning mode $2\,\mathrm{s}$, per each visited location and the camera travels at the speed of $1\,\mathrm{cm\,s^{-1}}$.

HyT_X_GMM proved to be statistically the best ranking method; on side 1, it was not possible to order Greedy, HistDayGrid, HyT_X_GMM, HyTted_GMM, WHyTe, time_window_GMM as the best ranking methods.

### B. Covering Metrics Results

Figure 4 presents $CN$ how many collection cycles and $MN$ how much traveled distance our methods needed on average to achieve the coverage $C = .99$ for various values of the allowed ratio of locations $a$. All results converge on $a = 1$ as there is no longer any selection, and the methods visit all locations.

As expected, the number of scans drops significantly with how many locations the robot is allowed to visit and plateaus around seven on side 0 and nine on side 1. The marginal benefit of visiting more locations also diminishes, which is caused by overlaps in the field of view of the camera between locations. There are clear, substantial differences between the uninformed methods and informed methods. When specifically optimizing for the coverage, the methods gain about $50\,\%$ edge for $a = .5$.

The amount of traversed meters is less intuitive. Only the Random method performs clearly worse as it selects locations randomly scattered around the whole area, therefore having to traverse up to twice the distance. For the lowest and highest $a$, there is almost no difference between

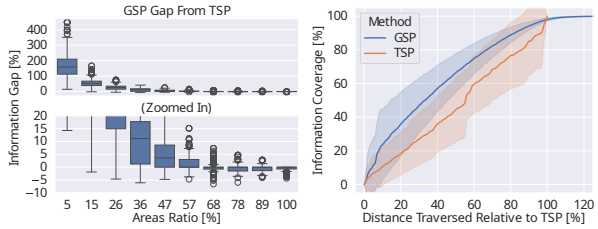


Fig. 6: Preliminary results on selection of locations under inspection (TSP) and searching paradigm (GSP). On the left, we compare for each $a$ the relative improvement of expected information $IG$ collected if the time budget would be given by our TSP solution over selected locations $L$. On the right, we compare information coverage $IC$ as the proportion of collected information to the distance traveled by our TSP solution.

Sequential and informed methods; for the middle values, the informed methods make better use of the overlapping views and information redundancy.

For both metrics, the performance of the informed methods is relatively similar. There is not a large practical difference between greedy and predictive methods. That would indicate that there is not a large benefit to be gained by the predictive selection.

### C. Real Time Results

To know the actual amount of time the robot needs to achieve the $C = .99$ coverage, we used specific values from our system and data from Fig. 4 to produce Fig. 5. The camera moves $1\,\mathrm{cm\,s^{-1}}$, and we consider two modes of data collection—observing and scanning. Observing mode stops at each location for $30\,\mathrm{s}$ before moving on to the next one. Scanning mode only takes one image, which results in an overhead of about $2\,\mathrm{s}$ caused by the need for stabilization, focusing, and exposition.

For the scanning mode, considering the data collection time makes no real difference compared to the traversed distance. Even on side 1 with 90 locations, the overhead for $a = 1$ only makes $3\,\mathrm{min}$. The best methods take close to an hour of operation time to construct the daily snapshot.

In the observation mode, the overheads are, however substantial and dominate other effects. Even though the number of scans necessary plateaus for the informed methods after $a = .5$, the amount of time grows. In terms of time, the fastest coverage would be achieved at only $a = .1$, taking between $3\,\mathrm{h}$ to $4\,\mathrm{h}$.

### D. Potential Improvements in Location Selection

Assessing the suboptimality of our location selection compared to the searching approach, the results in Fig. 6 (left) show that within the same travel budget, the GSP solution can collect substantially more relative expected information for $a < 50\%$, while for $a > 50\%$, the reported gap is not as prominent, usually less than $10\,\%$, and sometimes even attains negative values.

Assessing the effect of a possible unpredictable interruption of the data collection cycle, the results in Fig. 6 (right) show that the GSP-based approach, on average, can collect more information up to almost $100\%$ distance traversed by the TSP-based solution. However, around that point, the TSP solution becomes better because it is overall shorter, indicating that the TSP approach is more suitable for complete coverage, while the GSP approach is better when planning is done on an unknown budget.

## VII. DISCUSSION

In our experiments, informed methods proved to be an improvement over the current uninformed approach across a variety of covering metrics. For scanning mode, the improvement of needed time is not so prominent because the time of data collection is relatively small. However, once the data collection at each location takes more time—the observation mode—it becomes much more important to select locations with high information gain.

It is, however, harder to distinguish between the informed methods. In terms of the AUC, we see the predictive methods performing statistically better than uninformed and Greedy. This means that the MoDs developed for the human urban environments were able to learn some of the dynamics of the bee presence successfully. The small improvement in informed methods can be caused by only relatively small fluctuations in the number of bees in the hive. Nevertheless, an example visualization of the second cycle in Fig. 1 shows that the predictive method chose correctly the least bee-populated locations to visit. This suggests, that MoDs are gonna provide an edge for other applications in need of prediction of the bee density.

One of the important results we see is the effect of the areas ratio $a$ on the speed of coverage and that always visiting all locations is suboptimal. In terms of traveled distance, there is a sweet spot of $a = .5$ for which the ratio of gained information to the scatter of locations with high expected information is the largest. With the current setting of the system, scanning of the hive is done opportunistically when the queen rests, and a full scan is done every time. In the month-long dataset, the number of scans per day was, on average, 25 on side 0 and 38 on side 1, but in extremes, this went up to 56 and 112 scans in a day. Our results show that using informed methods, we would need only about ten scans, even with $a = .5$, to achieve appropriate coverage. That is only about $20\,\%$ of total collected data for the average day on side 0 and $13\,\%$ on side 1 (fewer scans and fewer locations), which is a substantial improvement should the robotic system be deployed in scale in the future.

We see that we were able to improve the effectiveness of data collection in terms of the number of visits, which is the dominant factor when the cost of data collection dominates the cost of travel—in the considered observation mode. However, for the scanning mode, the travel costs are more important, so including these costs in the selection of locations would likely allow for further improvements. This is confirmed in the comparison to the GSP method,

which was more efficient in planning for the collection of most expected information given the same budget. The limitation of this result is that we only estimate the expected information, and the collections of data are not independent because of overlaps of the camera's field-of-view.

## VIII. CONCLUSION

Our robotic system observes a living honeybee colony over long periods of time and balances data collection between the queen and the rest of the colony. In preparation for the new observing season, we studied the options for more effective observations of the comb, which need to happen daily and deal with occlusions by bees. We presented several methods that consider the dynamics of bee presence and can significantly lower the number of data-collecting cycles compared to our current setup. This will allow us to improve our system, lower the amount of low-information data, and help with the design of a bio-hybrid system.

In the future, we would want to investigate in more detail the capacities of our human-oriented MoDs to model the presence of bees. Moreover, there are several possible criteria for optimizing. We showed preliminary results on the searching paradigm and prepared the ground for a more thorough investigation of all options.

## ACKNOWLEDGMENT

This research was conducted under the European Union's Horizon 2020 research and innovation program grant "RoboRoyale" (agreement no. 964492). JM, MK and TK were funded by the European Union under the project Robotics and advanced industrial production (reg. no. CZ.02.01.01/00/22_008/0004590). Some authors were further supported by the Grant Agency of the Czech Technical University in Prague—JB, JU and TR by grant no. SGS22/168/OHK3/3T/13, JM by grant no. SGS23/175/OHK3/3T/13.

## REFERENCES

[1] R. Barmak *et al.*, "A robotic honeycomb for interaction with a honeybee colony," *Sci. Robot.*, vol. 8, no. 76, Mar. 2023, Art. no. eadd7385.

[2] D. Romano, M. Porfiri, P. Zahadat, and T. Schmickl, "Animal–robot interaction—an emerging field at the intersection of biology and robotics," *Bioinspiration Biomimetics*, vol. 19, no. 2, Feb. 2024, Art. no. 020201.

[3] W. Rajewicz *et al.*, "Organisms as sensors in biohybrid entities as a novel tool for in-field aquatic monitoring," *Bioinspiration Biomimetics*, vol. 19, no. 1, Nov. 2023, Art. no. 015001.

[4] O. M. Cliff, D. L. Saunders, and R. Fitch, "Robotic ecology: Tracking small dynamic animals with an autonomous aerial vehicle," *Sci. Robot.*, vol. 3, no. 23, p. Art. no. eaat8409, Oct. 2018.

[5] F. Angelini *et al.*, "Robotic monitoring of habitats: The natural intelligence approach," *IEEE Access*, vol. 11, pp. 72 575–72 591, Jul. 2023.

[6] M. Chellapurath, P. C. Khandelwal, and A. K. Schulz, "Bioinspired robots can foster nature conservation," *Frontiers Robot. AI*, vol. 10, Oct. 2023, Art. no. 1145798.

[7] M. Stefanec *et al.*, "A minimally invasive approach towards "ecosystem hacking" with honeybees," *Frontiers Robot. AI*, vol. 9, Apr. 2022, Art. no. 791921.

[8] J. E. Watson *et al.*, "Catastrophic declines in wilderness areas undermine global environment targets," *Current Biol.*, vol. 26, no. 21, pp. 2929–2934, Nov. 2016.

[9] C. A. Hallmann *et al.*, "More than 75 percent decline over 27 years in total flying insect biomass in protected areas," *PLOS ONE*, vol. 12, no. 10, Oct. 2017, Art. no. e0185809.

[10] J. Ulrich *et al.*, "Long-term tracking of individual, collective and social behaviors in honeybees by cooperating robots," *Sci. Robot.*, 2024, to appear.

[11] K. Žampachů *et al.*, "A vision-based system for social insect tracking," in *2022 2nd Int. Conf. Robot., Automat. Artif. Intell. (RAAI)*, Singapore, SG, Dec. 2022, pp. 277–283.

[12] T. P. Kucner *et al.*, "Survey of maps of dynamics for mobile robots," *Int. J. Robot. Res.*, vol. 42, no. 11, pp. 977–1006, Sep. 2023.

[13] T. Vintr *et al.*, "Toward benchmarking of long-term spatio-temporal maps of pedestrian flows for human-aware navigation," *Frontiers Robot. AI*, vol. 9, Jul. 2022, Art. no. 890013.

[14] F. Boenisch *et al.*, "Tracking all members of a honey bee colony over their lifetime using learned models of correspondence," *Frontiers Robot. AI*, vol. 5, Apr. 2018.

[15] K. Bozek *et al.*, "Markerless tracking of an entire honey bee colony," *Nature Communications*, vol. 12, no. 1, Mar. 2021.

[16] W. Bejjani *et al.*, "Occlusion-Aware Search for Object Retrieval in Clutter," in *2021 IEEE IROS*. Prague, Czech Republic: IEEE, Sept. 2021.

[17] J. Higgins and N. Bezzo, "A model predictive-based motion planning method for safe and agile traversal of unknown and occluding environments," in *2022 IEEE ICRA*. Philadelphia, PA, USA: IEEE, May 2022.

[18] M.-Y. Yu and and, "Occlusion-aware risk assessment for autonomous driving in urban environments," *IEEE RAL*, vol. 4, no. 2, Apr. 2019.

[19] P. Vansteenwegen *et al.*, "The orienteering problem: A survey," *Eur. J. Oper. Res.*, vol. 209, no. 1, 2011.

[20] E. Koutsoupias *et al.*, "Searching a fixed graph," in *Automata, Languages and Programming*, G. Goos, J. Hartmanis, J. Leeuwen, F. Meyer, and B. Monien, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 1996, vol. 1099, series Title: Lecture Notes in Computer Science.

[21] T. P. Kucner, A. J. Lilienthal, M. Magnusson, L. Palmieri, and C. S. Swaminathan, *Probabilistic Mapping of Spatial Motion Patterns for Mobile Robots*, 1st ed. Cham, CH: Springer, 2020.

[22] T. Vintr, "Maps of dynamics for social-aware navigation of autonomous mobile robots," Ph.D. dissertation, Czech Technical University in Prague, in review 2023.

[23] P. Biber *et al.*, "Dynamic maps for long-term operation of mobile service robots," in *Robotics: Science and Systems (RSS)*. Cambridge, MA, USA: Citeseer, 2005.

[24] T. Krajnik *et al.*, "Spectral analysis for long-term robotic mapping," in *2014 IEEE ICRA*. Hong Kong, CH: IEEE, 2014.

[25] J. P. Fentanes *et al.*, "Now or later? Predicting and maximising success of navigation actions from long-term experience," in *2015 IEEE ICRA*. Seattle, WA, USA: IEEE, 2015.

[26] T. Krajník, J. P. Fentanes, J. M. Santos, and T. Duckett, "Fremen: Frequency map enhancement for long-term mobile robot autonomy in changing environments," *IEEE Trans. Robot.*, vol. 33, no. 4, pp. 964–977, Aug. 2017.

[27] T. Krajník *et al.*, "Warped hypertime representations for long-term autonomy of mobile robots," *IEEE Robot. Autom. Lett.*, vol. 4, no. 4, pp. 3310–3317, Oct. 2019.

[28] G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics yolov8," 2023. [Online]. Available: https://github.com/ultralytics/ultralytics

[29] J. Vankerschaver and et al., "PyConcorde, v0.1," 2021. [Online]. Available: https://github.com/jvkersch/pyconcorde

[30] W. Cook, "Concorde," 2003.

[31] W. J. Conover, *Practical Nonparametric Statistics*, 3rd ed. New York, NY, USA: Wiley, 1999.

[32] M. Terpilowski, "scikit-posthocs: Pairwise multiple comparison tests in Python," *J. Open Source Softw.*, vol. 4, no. 36, Apr. 2019, Art. no. 1169.

[33] Z. Šidák, "Rectangular confidence regions for the means of multivariate normal distributions," *J. Amer. Statistical Assoc.*, vol. 62, no. 318, pp. 626–633, Jun. 1967.

[34] J. Mikula and M. Kulich, "Solving the traveling delivery person problem with limited computational time," *Central Eur. J. Oper. Res.*, vol. 30, no. 4, pp. 1451–1481, Dec. 2022.