

Track-a-Forager: a program for the automated analysis of RFID tracking data to reconstruct foraging behaviour

A. Van Geystelen¹ · K. Benaets¹ · D. C. de Graaf² · M. H. D. Larmuseau^{1,3} · T. Wenseleers¹

Received: 3 August 2015 / Revised: 18 November 2015 / Accepted: 19 November 2015 / Published online: 16 December 2015
© International Union for the Study of Social Insects (IUSSI) 2015

Abstract Behavioural studies make increasingly use of the passive radio-frequency identification (RFID) technology to monitor the foraging behaviour and activity patterns of individual animals over extended periods of time. Central place foragers, such as social insects, birds and many rodents have proved particularly well suited for this technology. As yet, however, there is no standardized methodology to filter and postprocess the data resulting from RFID scanners. Here we present a new user-friendly, publically available Java program named “Track-a-Forager” to analyse and rigorously filter RFID animal tracking data. The program is particularly suited and has special features to analyse social insect behaviour, but it is generic enough to analyse data obtained from any species. The implemented filtering algorithm consists of several well-defined steps to cluster multiple temporally clustered RFID scans of the same individual, determine events of leaving and entering the nest and/or feeder and reconstruct foraging trips for each individual. Track-a-Forager analyses RFID data independent of the used scanner system for eight

different types of standard experimental setups that are common in foraging behaviour research. These setups differ with respect to whether or not foraging at an artificial feeder is monitored and the specific placement of the RFID scanners at the nest or feeder. As a real-life example, we show how Track-a-Forager enables one to reconstruct 75 % more foraging trips compared to if one were to use the raw data.

Keywords Passive RFID · Software · Foraging behaviour · Social insects

Introduction

Radio-frequency identification (RFID) is a wireless sensor technology that can be utilised for the identification of goods, locations, animals, and even people (van Lieshout et al. 2007). In its most commonly used form, the active reader-passive tag (ARPT) system, the RFID tag transfers its identity to the reader upon being activated by radio signals or laser light (Fig. 1). There are three components in the ARPT system: a tag, an antenna and a reader which is connected to a computer (or data logger) to store the recorded scans (Scheiner et al. 2013). Typically, the antenna and reader are packed into a larger structure called the scanner that controls the data communication of the system (Kissling et al. 2013). When laser light is used to activate the tag, the light is detected by the photocell in the tag and this gives the antenna of the tag the energy to emit radio signals specific for each tag such that the information on the tag’s microchip can be read. When the radio signals emitted by the scanner’s antenna at a specific frequency are used to activate the tag, the tag backscatters the radio signals with a modulated frequency such that the information on the tag’s microchip can be read. In the case of passive tags, the power

Electronic supplementary material The online version of this article (doi:10.1007/s00040-015-0453-z) contains supplementary material, which is available to authorized users.

✉ A. Van Geystelen
anneleen.vangeystelen@bio.kuleuven.be

- ¹ Laboratory of Socio-ecology and Social Evolution, Department of Biology, Katholieke Universiteit Leuven, Naamsestraat 59, 3000 Leuven, Belgium
- ² Molecular Entomology and Bee Pathology, Department of Biochemistry and Microbiology, Ghent University, Ghent, Belgium
- ³ Forensic Medicine, Department of Imaging and Pathology, Leuven, KU Leuven, Leuven, Belgium

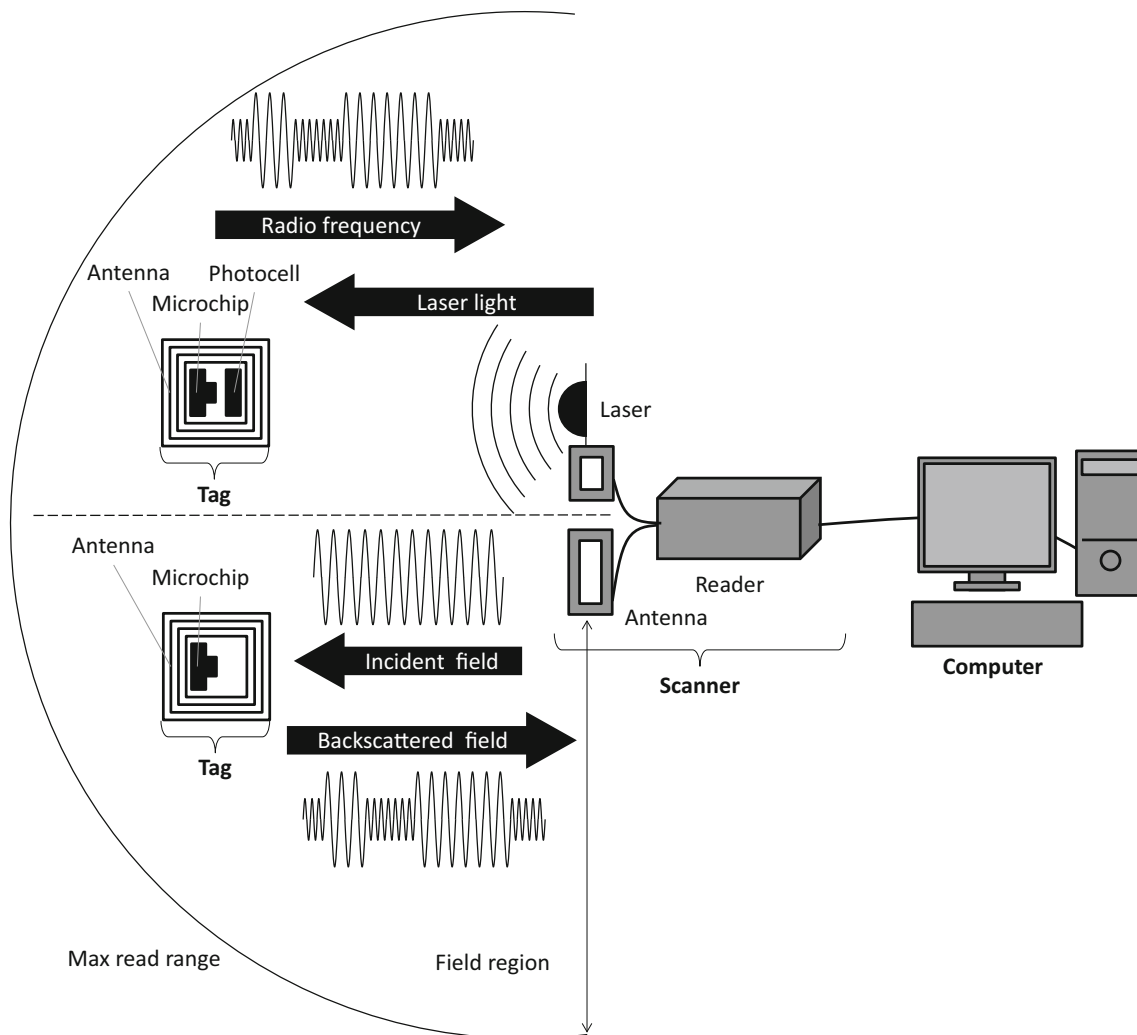


Fig. 1 Design of the passive RFID system. There are two different ways in which the tag can be activated: via laser light (*above the dashed line*) or via radio signals (*under the dashed line*). When laser light is used to activate the tag, the light is detected by the photocell and give the antenna of the tag the energy to emit radio signals specific for each tag such that the information on the tag's microchip can be read. When the radio signals emitted by the scanner's antenna at a specific frequency are used to activate the tag, the tag backscatters the

radio signals with a modulated frequency such that the information on the tag's microchip can be read. For each tag the radio signals emitted/backscattered will be different such that the tag can be identified based on those radiosignals. It is the reader that will decode the signals and send the data consisting of tag ID, scanner ID and time stamp to the computer. The tag only gets activated when it is enters the field region of the scanner's antenna, i.e. when the distance between the antenna's is less than the max read range

enabling of the tag to communicate with the reader is drawn from the movement across the electromagnetic field of the antenna or from the light activation of the tag's photocell, which means that no on-board batteries are required.

Passive RFID technology has become a popular tool for tracking the foraging behaviour and movement of small wildlife animals due to its ability to individually identify each free-living animal without disturbing the animals' movements (Robinson et al. 2009; Streit et al. 2003). The use of small and light passive RFID tags that do not require any on-board batteries are particularly attractive as their small size enables behavioural data to be collected with minimal bias, and without any human interference

(Hou et al. 2015). Furthermore, it is a solution for behavioural research which requires precise and long-term observations of many individuals that are difficult or unfeasible to obtain by direct human observation (Kurazono et al. 2013).

As the scanner is stationary, RFID technology is most frequently used to track the foraging activity of central place foraging species and log the times of entering and leaving the burrow or nest (Pinter-Wollman and Mabry 2010), e.g. in birds (Kurazono et al. 2013; Naumowicz et al. 2008; Sales et al. 2015), social insects (Henry et al. 2012; Robinson et al. 2012; Stelzer et al. 2010; Tenczar et al. 2014) or small rodents (Scheibler et al. 2013; Scheibler et al.

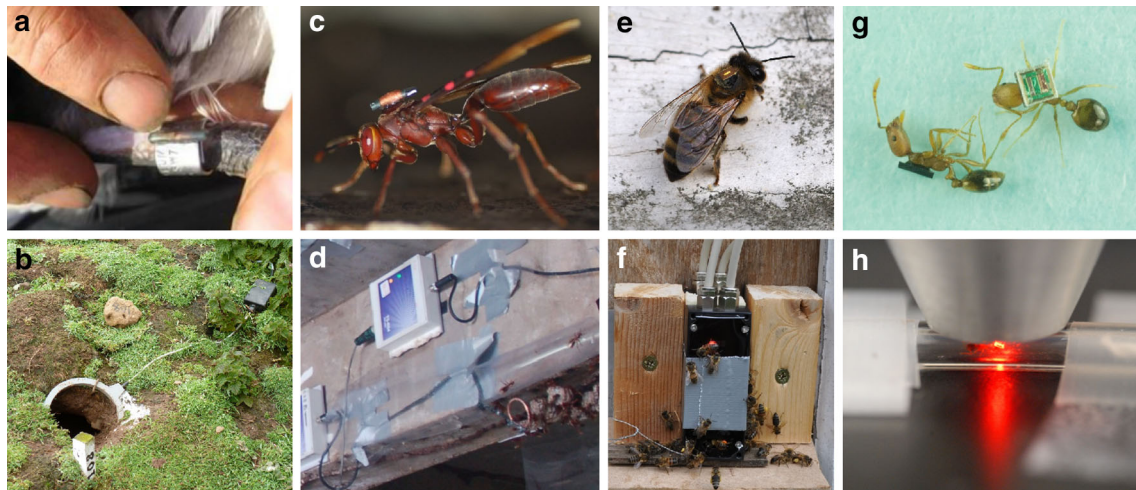


Fig. 2 Photos of RFID tag and the corresponding antenna and reader used with different sized animals: **a, b** *Puffinus puffinus* (copyright Robin Freeman, University of Oxford), **c, d** *Polistes canadensis* (copyright Patrick Kennedy, University of Bristol), **e, f** *Apis mellifera* (copyright Kristof Benaets, KU Leuven), **g** *Monomorium pharaonis*

(copyright Phil Roberts, University of York) and **h** *Temnothorax albipennis* (copyright Elva Robinson, University of York). In *Monomorium pharaonis* (**g**) and *Temnothorax albipennis* (**h**) the RFID tags are activated by light instead of radio signals

2014; Serra et al. 2012). In both ornithological and social insect research (Fig. 2), RFID tracking has also been used to study foraging performance by placing RFID scanners at both the nest and feeding stations (Bonter and Bridge 2011; Bonter et al. 2013; Decourtye et al. 2011; Henry et al. 2012; Hou et al. 2015), e.g. enabling researchers to measure the impact of pesticides on the foraging performance of honeybees (Decourtye et al. 2011; Henry et al. 2012; Gill and Raine 2014; Schneider et al. 2012). Finally, complex setups that combine several RFID scanners at fixed places have also been used to spatially monitor European badgers (Noonan et al. 2014), Norwegian lobsters (Aguzzi et al. 2011) and mice (Weissbrod et al. 2013), as well as nest-drifting and intraspecific reproductive parasitism in tropical paper wasps (Sumner et al. 2007) and stingless bees (Van Oystaeyen et al. 2013).

Although the benefits of using RFID technology to log foraging activity of small animals is clear, errors in the raw RFID data can significantly complicate data analysis, and extensive filtering is required before data can be interpreted. Two problems that routinely arise in any RFID platform are rapid-succession scans of the same individual and missed scans. Rapid-succession scans are successive scans of the same individual by the same scanner within a small time range, and are one of the most common types of problems, which can be caused by the lingering of the animal underneath the scanner or the animal moving around the nest entrance (Scheiner et al. 2013). Rapid-succession scans are a particularly severe problem in RFID-based studies of social insects (Scheiner et al. 2013; Tenczar et al. 2014), as their large colony sizes typically result in queues at the nest entrance or exit, and in many workers, e.g. guards, moving

in and out of the nest in quick succession. A second major problem can arise when some passages are missed by the scanners (“missed scans”), as without adequate filtering, this could lead to the erroneous inference of prolonged stays outside the nest. Missed scans can arise when the distance between the tag and scanner is too large (Kissling et al. 2013; Pinter-Wollman and Mabry 2010; Scheiner et al. 2013) or when the animal passes at a suboptimal angle in the scanner tunnel (Scheiner et al. 2013). Indeed, detection distance becomes a major constraint especially in the highly miniaturized passive RFID tags used to study small insect species, as there is a trade-off between detection distance and the size of the tag. Hence, missed scans occur most commonly and are particularly problematic in RFID studies of small animal species, such as social insects (ants, bees or wasps). Tags used for these kinds of animals are typically around 1–2 mm² large, but can be as small as 0.5 mm² in size, resulting in a detection range that is limited to a few millimetres or less, and causing missed scans to be common (Hou et al. 2015).

Previously, only few methods to apply standardized data filtering to RFID tracking data have been developed to analyse RFID data for studies on animal behaviour. These existing interfaces, however, only handle data of one particular scanner system, are closed source, dependent on particular networks of scanners to be able to analyse spatial patterns or/and work as a black box, meaning that there are no published details of their internal algorithm. Example of such previously developed systems are IntelliCage (Richardson 2012), a Web application by Catarinucci et al. (2014), a LabVIEW application (Aguzzi et al. 2011), Beegroup DB2Use (Streit et al. 2003), rfbec (www.nspyre.nl/)

rfibee) and TimeBee[®] (CTIS, rillieux-la-pape, France; Devillers 2014). Recently, standardised methods for the use of RFID systems in animal behavioural research were published in the COLOSS BEEBOOK (Scheiner et al. 2013). Nevertheless, given that the authors only present a vague description of possible filtering algorithms and advise to work with custom-made scripts, it is clear that there is an urgent need for publically available software with well-described algorithms to adequately analyse RFID tracking-based behavioural data.

This study presents a user-friendly, publically available, clear box Java program called “Track-a-Forager” for the analysis and filtering of RFID-based behavioural data. As missed scans are major problems occurring in RFID data, the algorithm of Track-a-Forager deals with them by also allowing incomplete sequences of scans as foraging trips. For each type of setup, these incomplete sequences and their details will be different. Eight different types of standard experimental setups (Fig. 3) are the particular strength of the software and have been used in studies of foraging behaviour and intraspecific social parasitism (Abou-Shaara 2014; Gill et al. 2012; Meikle and Holst 2014; Seeley 1995; Sumner et al. 2007; Van Oystaeyen et al. 2013). Furthermore, the application domain of the program is not limited to small insects, such as bees, ants or wasps, for which error rates of existing RFID systems tend to be high and extensive data filtering is required, but can equally be used for the analysis of RFID data of larger animals, such as small mammals, fish or birds, for which fewer artefacts are expected. In our manuscript, however, we will use social insects as a standard example, because of the inherent complexity in analysing RFID-data in such systems.

Track-a-Forager

The different standard experimental setups supported by Track-a-Forager differ in the way in which foraging behaviour is monitored and in the number and position of RFID scanners at the nest and/or feeder (Fig. 3). In particular, we make a distinction between setups where only the leaving and entering the nest by each individual is monitored (“natural foraging”, upper panel in Fig. 3) and setups where the nest scanners are complemented with a scanner at an artificial feeder (“artificial foraging”, lower panel in Fig. 3). Setups with monitoring at the feeder have more details about the foraging trip and, therefore, tolerate more missed scans when reconstructing the foraging trips. In addition, we make a distinction between situations where the nest entrance and exit are separated using one-way tunnels and those where the nest entrance and exit are shared (Fig. 3), and between situations where one or two scanners are placed at each nest entrance, nest exit or feeder (Fig. 3). The use of two RFID scanners enables one to infer the direction of motion of the animal and distinguish events of leaving or entering the nest hereby more missed scans can be tolerated when reconstructing the foraging trips. In setups where the entrance and exit of the nest are separated by one-way tunnels, the entrance and exit, however, can in principle each be monitored by a single scanner. Similarly, for applications where no inferences about the direction of movement are required, e.g. in some social parasitism experiments or in strictly controlled environments with tunnels from nest to feeder, only one scanner per nest opening can be sufficient. However, by using two scanners more data points are added to the analysis which enables

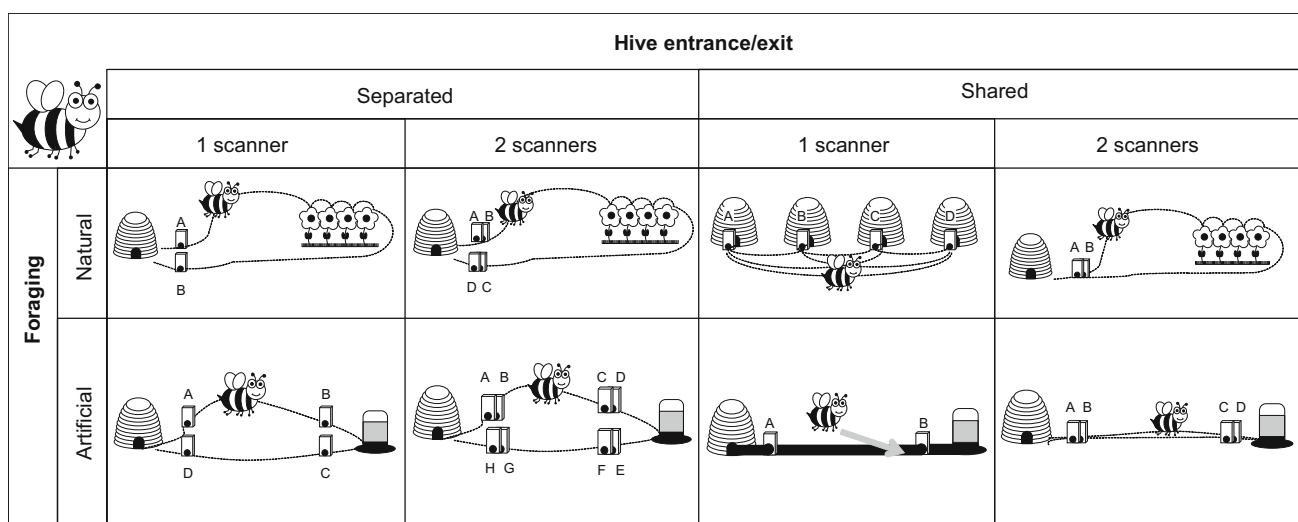


Fig. 3 Overview of the eight experimental setups that can be analysed by the Track-a-Forager software. They differ in the monitoring of the foraging behaviour which can be natural (only scanners at nest) or

artificial (scanners at nest and artificial feeder), the nest entrance/exit (separated or shared) and in the number of RFID scanners at the nest and feeder (one or two)

Track-a-Forager to tolerate missed scans when reconstructing the foraging trips. To avoid interference between the two scanners, they should be placed in series at a distance recommended by the manufacturer of the scanner system.

Algorithm

Track-a-Forager has a clear, well-defined algorithm to filter the raw data of scans. Each scan in the RFID dataset comprises three types of data: the unique identifier of the tag, the label of the scanner that retrieved the tag identifier, and the time point the tag was detected. The Track-a-Forager algorithm, which handles the RFID data filtering, is divided into three stages (Fig. 4a), with user-defined filtering steps being applied at each stage. In the first stage, the raw, chronologically ordered scans are filtered to cluster the rapid-succession scans via a sliding window approach based on all three available data points (Fig. 5). This is done by comparing each scan of a unique tag registered by a specific scanner (currentScan), and clustering the two if the time difference between them is smaller than a user-defined cut-off value. This results in a number of unique clusters of scans representing a unique passage of an individual at any one of the scanners in the setup. The default value for the user-defined cut-off to cluster scans is 20 s and we don't recommend to set it lower since scans will be treated as separated clusters. When there are enough tagged foraging bees, the distinction between the clusters will mainly be determined by the sequence of the transponder IDs and less by the time points of the scans.

Once rapid-succession scans are clustered together, a second processing step—also based on a sliding window approach—is used to detect events of leaving (OUT) or entering (IN) the nest and/or feeder in the setups with two scanners (Fig. 4b) or with separated nest entrance and exit

(Fig. S1). For setups with a single scanner at a shared nest entrance and exit, this processing step is not applied; as such setups do not allow one to discriminate between OUT and IN events. The idea of this stage of the analysis is to use information on the direction of motion to annotate successive scan clusters at paired scanners as either an OUT or an IN event, and to do this only if the time difference between the scan clusters is smaller than a user-defined cut-off value. As missed scans can occasionally occur, there are also scan clusters which cannot be classified as OUT or IN events in the setups with paired scanners. These unclassified scan clusters; however, can still be useful to detect the occurrence of foraging trips. Subsequently, in setups with one or two scanners placed at an artificial feeder, we also use a sliding window approach to annotate events of going from the nest to the feeder (GO) or from the feeder to the nest (RETURN) based on classified OUT and IN events as well as any unclassified scan clusters (Fig. S1). The default value for the user-defined IN–OUT cut-off is 20 s but we recommend to change it according to distance between the two scanners. For example, it takes an individual more time to travel between scanners placed 20 cm apart than when the distance between the scanners is 4 cm. Also the number of foragers in the colony can have an effect on the time necessary to move from one scanner to the other as queues can occur when a lot of foragers want to enter the nest at the same time.

In the third and last stage of the algorithm, the order of the OUT, IN, GO, and RETURN events is used to reconstruct foraging trips (Fig. S2), again based on a sliding window approach. In setups that do not involve scanners at feeders, these foraging trips are reconstructed solely based on OUT and IN events. By contrast, in setups with two scanners, we also use information from unclassified scan clusters. Figure S1 gives an overview how we infer OUT, IN, GO and RETURN events from the scan order and how they are used to reconstruct foraging trips. Finally, to

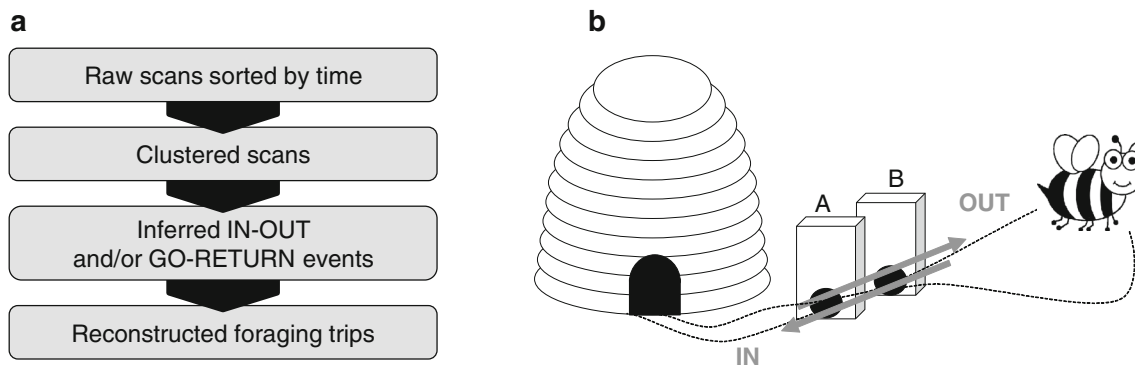
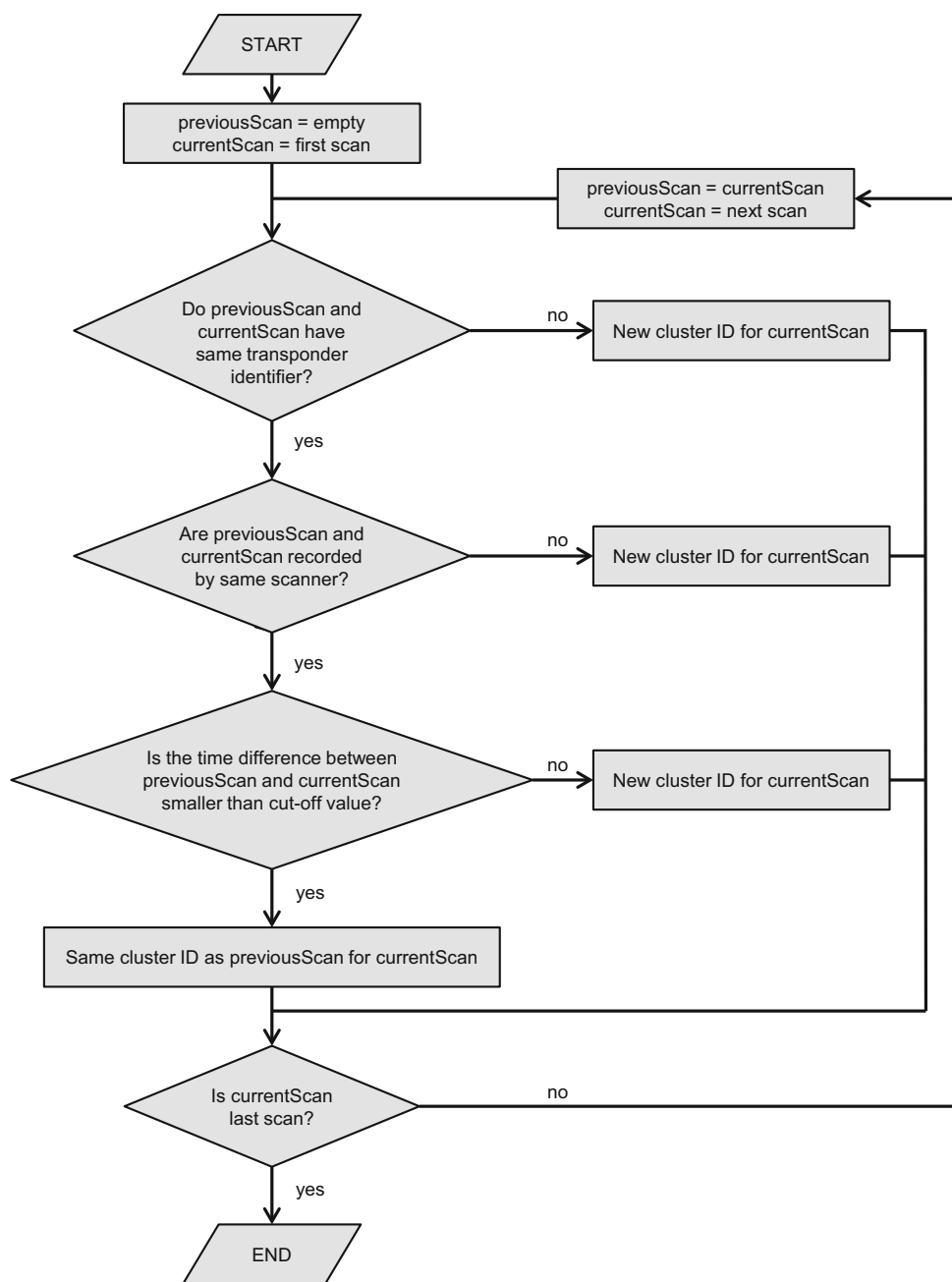


Fig. 4 **a** The general three-step Track-a-Forager algorithm for filtering the RFID scan data which leads to reconstructed foraging trips for the eight experimental setups. **b** The inferring of the OUT and

IN events in setups with two scanners at the nest entrance/exit and/or the feeder depends on the order of the scanners: the OUT event has the order AB while the IN event has a BA order

Fig. 5 Flowchart of the first step of the Track-a-Forager algorithm: clustering of the scans to treat rapid-succession scans (of the same tag identifier made by the same scanner) as one scanning event when these scans occur within a certain, user-defined timespan (cut-off value which can be given in the graphical user interface). By comparing each scan (currentScan) with the previous (previousScan) scan a sliding window approach is used



remove short-duration stays of the animals outside the nest (e.g. due to the movement of guard bees in honeybee studies) and to remove trips of exceptionally long duration, the user can specify a minimal threshold and a maximal cut-off value for the length of each foraging trip. The default minimal foraging trip length threshold is 300 s (i.e. 5 min) to avoid identifying the movement of guards as foraging trips. The default maximal foraging trip length cut-off is 86,400 s (i.e. 24 h) but this includes overnight stays outside the hive. This can be overcome by lowering the cut-off value depending on the hours of daylight during the experiment or by selecting the foraging trips in the output

file on the indication whether or not the foraging trip spanned more than one day.

As Track-a-Forager is a ready-to-use software program several additional output options are implemented to make it as user-friendly as possible. These options include exporting the foraging frequency per individual over time and exporting the detailed foraging durations and the “age” relative to the start date of the experiment of each individual at the first and last scan and trip. For maximum ease of operation, any annotation information, e.g. any treatments which were given to particular sets of individuals, the colony to which each individual belongs, its age at the time of

starting the experiment, etc., provided by the user is also integrated in these output files.

Real-life example

To demonstrate the usefulness of the Track-a-Forager software in behavioural research and to compare several values for the parameters in the algorithm, a real-life example is provided. The RFID data used in this real-life example came from observations of *Apis mellifera carnica* honeybees that were kept in a 3-frame observation hive at the laboratory's apiary in Leuven, Belgium. The host colony contained two frames of brood, one frame with stored pollen and honey, a queen and around 3000 host colony workers and was placed indoors at room temperature and were connected to the outside via a single entrance tunnel to allow free foraging. At the end of the tunnel two iID[®] MAJA 4.1 RFID scanner modules (Microsensys, Germany) were placed in series, which were connected to a MAJA 4.1 host computer (Microsensys, Germany) to record and log the timing of all RFID tagged honeybees leaving or entering the hive. The scanners were separated from each other by a 4 cm wooden tunnel block to prevent interference between the scanners. Bees were allowed to emerge by placing brood frames in a MIR-253 incubator (Sanyo, Belgium) at 34 °C and 60 % humidity, after which newly eclosed workers were collected daily. A total of 400 bees were tagged with a mic3[®] 64-bit read-only RFID transponder (Microsensys, Germany) by gluing the tag to the bee's thorax using Kombi Turbo two-component glue (Bison, Netherlands). The tags measured 2.0 × 1.7 × 0.5 mm, weighed less than 5 mg and transmitted at 13.56 MHz. The RFID codes of all tagged workers, together with time of introduction, were added to a transponder information database by reading each code using the iID[®] PENmini USB pen. Up to 50 tagged individuals were kept in 15 × 10 × 7 cm cages kept at 34 °C and 60 % humidity, and contained a 10 × 8 cm piece of honey-filled comb and drinking water, to allow the bees to settle down before introducing them into the host colonies. Before introduction, the cages were placed on top of the observation hives, separated only by a wire mesh, for a 30 min period to increase acceptance rates. The workers were introduced over the course of a period of 5 days, and foraging behaviour was monitored from 1 August until 26 August 2012 whereby there were on average 14.64 h of daylight.

Track-a-Forager was used to analyse the raw data, using the setup of natural foraging with two adjacent scanners at a shared hive entrance/exit (top right panel in Fig. 3) and using the default time constraint parameters. The raw data consisted of almost 44,000 raw scans of 281 bees. In the first processing step, rapid-succession scans were eliminated,

resulting in the grouping of the raw scan into 17,117 scan clusters. In the second processing step, 1561 leaving (OUT) and 1968 entering (IN) events were detected, whereas the remaining scan clusters stayed unclassified. Finally, in the last stage, a total of 2101 foraging trips of 183 bees were reconstructed of which 488 were complete, being based on a successive OUT and IN event pair. The incomplete reconstructed foraging trips which are based on an unclassified scan cluster and an IN or OUT event show that relatively more OUT events were missed since 894 trips miss an OUT event while 719 trips miss an IN event. Overall, in this dataset, fewer than 25 % of the reconstructed trips were complete. This means that the use of Track-a-Forager resulted in a much more complete analysis than would have been possible based on simpler manual analysis of the data. The bees were on average 9.57 ± 1.61 days (mean \pm SD) old on their first foraging trip and spend on average 1.00 ± 1.04 h (mean \pm SD) outside the hive.

Beside running Track-a-Forager with the default parameters (i.e. 20 s for the cluster cut-off, 20 s for the IN–OUT cut-off, 300 s for the flight minimal threshold and 86,400 s for the flight maximal cut-off), we also ran it with no limitations regarding the minimal and maximal flight lengths while the default values for the cluster and IN–OUT cut-off remain. In a third analysis no restriction is put on the IN–OUT cut-off which means that any consecutive clusters of scans of the same RFID made by two different scanners will be considered as IN/OUT event regardless the time difference between the clusters. The last analyses are done with a larger (35 s) and smaller (5 s) value for the cluster cut-off while using the default values for the other parameters. As expected there are more reconstructed trips when there are no restrictions on the trip length (Fig. S3). The need to filter for guarding bees is shown by the fact that there are much more reconstructed trips with a length smaller than 300 s than there are with a length larger than 86,400 s, i.e. 498 trips compared to 17. When there is no time constraint on the determination of the IN/OUT events, the number of reconstructing foraging trip is lower than with the default IN–OUT cut-off value of 20 s (Fig. 3). One might expect the opposite, i.e. more reconstructed trips, however, this is explained by the fact that if there is no time restriction on which cluster of scans to group in an IN/OUT event, 'wrongly' grouped clusters have an influence on the consecutive clusters. For example, the time differences between four consecutive clusters A, B, C and D are 830, 5 and 37 s. Clusters A and C come from scanner 1 while clusters B and D are from scanner 2. Under default setting, cluster A and D would be an UNKNOWN event while the group of clusters B and C is an IN event. Therefore, cluster A and the group of clusters B and C would be a reconstructed foraging trip. When using no IN–OUT cut-off, clusters A and B are grouped in an OUT event while also

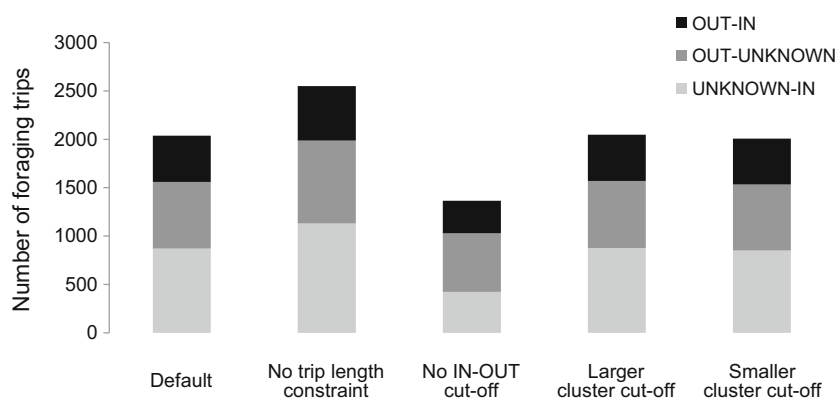


Fig. 6 Number of the reconstructed foraging trips for each analysis using different values for the parameters. The *black* indicates the foraging trips with a complete sequence of scans, i.e. both OUT and IN events consist of two clusters of scans. The *dark grey colour* points out that there was only one cluster of scans for the IN event while the light grey indicates the opposite, i.e. there was only one cluster of scans for the OUT event. The default setting are 20 s for the cluster cut-off, 20 s for the IN–OUT cut-off, 300 s for the flight minimal threshold and 86,400 s for the flight maximal cut-off. In the analysis ‘No trip length

constraint’ the limitations regarding the minimal and maximal flight lengths are omitted while the default values for the cluster and IN–OUT cut-off remain. In the third analysis no restriction is put on the IN–OUT cut-off which means that any consecutive clusters of scans of the same RFID made by two different scanners will be considered as IN/OUT event regardless the time difference between the clusters. The last analyses are done with a larger (35 s) and smaller (5 s) value for the cluster cut-off while using the default values for the other parameters

clusters C and D are grouped in an OUT event. Hereby no foraging trip can be reconstructed since there are both OUT events. Changing the cluster cut-off to a larger or smaller values does not have a significant impact on the number of reconstructed trips (Fig. 6) as the distinction between the clusters will mainly be determined by the sequence of the transponder IDs and less by the time points of the scans when there are enough tagged foraging bees. It seems that the IN–OUT cut-off, flight minimal length threshold and maximal length cut-off have a larger effect on the number reconstructed foraging trips than the cluster cut-off (Fig. 6, Fig. S4 and Fig. S5). The different analyses show that the used parameters in the Track-a-Forager algorithm are useful and that their default values are appropriate.

Information to download the Track-a-Forager software

Track-a-Forager can be downloaded as a ready-to-use Java interface with a detailed manual from our website: <https://perswww.kuleuven.be/~u0072398/>. The Java code itself is available upon request from the corresponding author.

Acknowledgments Kristof Benaets has a Ph.D.-scholarship of the Flemish Agency for Innovation by Science and Technology (IWT-Vlaanderen), Maarten H.D. Larmuseau is a postdoctoral fellow of the Research Foundation-Flanders (FWO-Vlaanderen). This study was funded the KU Leuven BOF—Centre of Excellence Financing on ‘Eco- and socio-evolutionary dynamics’ (Project number PF/2010/07) and a grant from the Research Foundation-Flanders (FWO-Vlaanderen, Grant G.0628.11 N). We thank Jelle van Zweden, Edgar Dueñez-Guzmán, Eliseo Ferrante and Dries Cardoen for useful

technical assistance and discussions and Robin Freeman, Patrick Kennedy, Phil Roberts and Elva Robinson for the photos.

References

- Abou-Shaara HF (2014) The foraging behaviour of honey bees, *Apis mellifera*: a review. *Vet Med* 59:1–10
- Aguzzi J, Sbragaglia V, Sarriá D, García JA, Costa C, del Río J, Mánuel A, Menesatti P, Sardà F (2011) A new laboratory radio frequency identification (RFID) system for behavioural tracking of marine organisms. *Sensors* 11:9532–9548
- Bonter DN, Bridge ES (2011) Applications of radio frequency identification (RFID) in ornithological research: a review. *J Field Ornithol* 82:1–10
- Bonter DN, Zuckerberg B, Sedgwick CW, Hochachka WM (2013) Daily foraging patterns in free-living birds: exploring the predation–starvation trade-off. *Proc Biol Sci* 280:20123087
- Catarinucci L, Colella R, Mainetti L, Patrono L, Pieretti S, Sergi I, Tarricone L (2014) Smart RFID antenna system for indoor tracking and behavior analysis of small animals in colony cages. *IEEE Sens J* 14:1198–1206
- Decourtye A, Devillers J, Aupinel P, Brun F, Bagnis C, Fourrier J, Gauthier M (2011) Honeybee tracking with microchips: a new methodology to measure the effects of pesticides. *Ecotoxicology* 20:429–437
- Devillers J (2014) *In Silico* Bees. CRC Press, Boca Raton
- Gill RJ, Raine NE (2014) Chronic impairment of bumblebee natural foraging behaviour induced by sublethal pesticide exposure. *Funct Ecol* 28:1459–1471
- Gill RJ, Ramos-Rodriguez O, Raine NE (2012) Combined pesticide exposure severely affects individual- and colony-level traits in bees. *Nature* 491:105–108
- Henry M, Béguin M, Requier F, Rollin O, Odoux JF, Aupinel P, Aptel J, Tchamitchian S, Decourtye A (2012) A common pesticide decreases foraging success and survival in honey bees. *Science* 336:348–350

- Hou L, Verdirame M, Welch KC (2015) Automated tracking of wild hummingbird mass and energetics over multiple time scales using radio frequency identification (RFID) technology. *J Avian Biol* 46:1–8
- Kissling DW, Pattemore DE, Hagen M (2013) Challenges and prospects in the telemetry of insects. *Biol Rev Camb Philos Soc* 89:511–530
- Kurazono H, Yamamoto H, Yamamoto M, Nakamura K, Yamazaki K (2013) RFID and ZigBee sensor network for ecology observation of seabirds. In: Proceedings of the 15th international conference on advanced communication technology (ICACT), 2013, pp 211–215
- Meikle WG, Holst N (2014) Application of continuous monitoring of honeybee colonies. *Apidologie* 46:10–22
- Naumowicz T, Freeman R, Heil A, Calsyn M, Hellmich E, Brändle A, Guilford T, Schiller J (2008) Autonomous Monitoring of vulnerable habitats using a wireless sensor network. In: Proceedings of the 3rd workshop on real-world wireless sensor networks (REALWSN 2008) 2008, pp 51–55
- Noonan MJ, Markham A, Newman C, Trigoni N, Buesching CD, Ellwood SA, Macdonald DW (2014) Climate and the Individual: inter-Annual Variation in the Autumnal Activity of the European Badger (*Meles meles*). *PLoS One* 9:e83156
- Pinter-Wollman N, Mabry KE (2010) Remote-Sensing of Behavior. In: Breed MD, Moore J (eds) *Encyclopedia of Animal Behavior*, 3rd edn. Academic Press, Oxford, pp 33–40
- Richardson CA (2012) Automated homecage behavioural analysis and the implementation of the three Rs in research involving mice. *Perspect Lab Anim Sci* 40:7–9
- Robinson EJM, Richardson TO, Sendova-Franks AB, Feinerman O, Franks NR (2009) Radio tagging reveals the roles of corpulence, experience and social information in ant decision making. *Behav Ecol Sociobiol* 63:627–636
- Robinson EJM, Feinerman O, Franks NR (2012) Experience, corpulence and decision making in ant foraging. *J Exp Biol* 215:2653–2659
- Sales GT, Green AR, Gates RS, Brown-Brandl TM, Eigenberg RA (2015) Quantifying detection performance of a passive low-frequency RFID system in an environmental preference chamber for laying hens. *Comput Electron Agr* 114:261–268
- Scheibler E, Roschlau C, Brodbeck D (2014) Lunar and temperature effects on activity of free-living desert hamsters (*Phodopus roborovskii*, Satunin 1903). *Int J Biometeorol* 58:1769–1778
- Scheibler E, Wollnik F, Brodbeck D, Hummel E, Yuan S, Zhang FS, Zhang XD, Fu HP, Wu XD (2013) Species composition and interspecific behavior affects activity pattern of free-living desert hamsters in the Alashan Desert. *J Mammal* 94:448–458
- Scheiner R, Abramson CI, Brodschneider R et al (2013) Standard methods for behavioural studies of *Apis mellifera*. *J Apicult Res* 52:1–58
- Schneider CW, Tautz J, Grünewalt B, Fuchs S (2012) RFID Tracking of Sublethal Effects of Two Neonicotinoid Insecticides on the Foraging Behavior of *Apis mellifera*. *PLoS ONE* 7:e30023
- Seeley TD (1995) *The Wisdom of the Hive: The Social Physiology of Honey Bee Colonies*. Harvard University Press, Cambridge
- Serra J, Hurtado MJ, Le Négrate A, Féron C, Nowak R, Gouat P (2012) Behavioral differentiation during collective building in wild mice *Mus spicilegus*. *Behav Process* 89:292–298
- Stelzer RJ, Chittka L, Carlton M, Ings TC (2010) Winter active bumblebees (*Bombus terrestris*) achieve high foraging rates in Urban Britain. *PLoS One* 5:e9559
- Streit S, Bock F, Pirk CWW, Tautz J (2003) Automatic life-long monitoring of individual insect behaviour now possible. *Zoology* 106:169–171
- Sumner S, Lucas E, Barker J, Isaac NN (2007) Radio-tagging technology reveals extreme nest-drifting behavior in a eusocial insect. *Curr Biol* 17:140–145
- Tenczar P, Lutz CC, Rao VD, Goldenfeld N, Robinson GE (2014) Automated monitoring reveals extreme interindividual variation and plasticity in honeybee foraging activity levels. *Anim Behav* 95:41–48
- van Lieshout M, Grossi L, Spinelli G, Helmus S, Kool L, Pennings L, Stap R, Veugen T, van der Waaij B, Boreau C (2007) RFID technologies: emerging issues, challenges and policy options. IPTS, Sevilla
- Van Oystaeyen A, Alves DA, Oliveira RC, Nascimento DL, Nascimento FS, Wenseleers T (2013) Sneaky queens in *Melipona* bees selectively detect and infiltrate queenless colonies. *Anim Behav* 86:603–609
- Weissbrod A, Shapiro A, Vasserman G, Edry L, Dayan M, Yitzhaky A, Hertzberg L, Feinerman O, Kimchi T (2013) Automated long-term tracking and social behavioural phenotyping of animal colonies within a semi-natural environment. *Nat Commun*. doi:10.1038/ncomms3018