sftraj: A central class for tracking and movement data

Submitted to: R Consortium's ISC

Mathieu Basille

2019-04-28

Signatories

The **Project team** lists the core members of the work, who will be instrumental in progress and completion of the project. **Contributors** actively helped with this proposal, and **Consulted** have been given the opportunity to provide feedback. Signatories are listed by alphabetical order. **Note: relevant R packages to which people contributed are indicated within brackets.**

Project team

- **Mathieu Basille**, Assistant Professor at the University of Florida, USA [`hab`, `rpostgis`, `rpostgisLT`]
- **Matt Boone**, Data Scientist at the University of Florida, USA
- **Clément Calenge**, Statistical Analyst at the *Office national de la chasse et de la faune sauvage*, France [`adehabitatHR`, `adehabitatLT`]
- **Rocío Joo**, Postdoctoral Associate at the University of Florida, USA
- **Emiel van Loon**, Assistant Professor at the University of Amsterdam, the Netherlands.

Contributors

- **Balázs Dukai**, Delft University of Technology, the Netherlands [`rpostgisLT`]
- **Ioannis Kosmidis**, University of Warwick, UK [`trackeR`]
- **Bart Kranstauber**, University of Zurich, Switzerland [`move`]
- **Edzer Pebesma**, University of Munster, Germany[`sf`, `spacetime` and `trajectories`]

Consulted

- **Francesca Cagnacci**, Edmund Mach Foundation, Italy
- **Hamish Campbell**, Charles Darwin University, Australia [`VTrack`]
- **Ross Dwyer**, University of Queensland, Australia [`Digiroo2`, `VTrack`]
- **Brett McClintock**, University of St. Andrews, Scotland [`moveHMM`, `momentuHMM`]
- **Théo Michelot**, University of St. Andrews, Scotland [`moveHMM`, `momentuHMM`]
- **Henning Teickner**, University of Munster, Germany
- **Vinay Udyawer**, Australian Institute of Marine Science, Australia [`VTrack`]
- **Ferdinando Urbano**, Freelancer, Italy

From the R Consortium:

- **Hadley Wickham**

The Problem

Movement defined broadly plays a central role in fields as diverse as transportation, sport, ecology, music, medicine, and data science (Gudmundsson *et al.* 2012). As a matter of fact, thanks to global navigation satellite system (GNSS) and radio-frequency identification (RFID), miniaturized tracking devices have become nearly ubiquitous, and resulted in an ever-increasing volume of localization data from various moving objects, such as humans or animals; cars, boats or planes; the hand of a musician playing a violin in 3 dimensions; the entire movement of an orchestra or a flock of starlings; or the eye of a person that changes its focus from one object to the other. Sampling all these movements results in the same type of data called *tracking data*, in the form of geographic (x, y, z) and temporal coordinates (t) (Joo *et al.* 2018). Despite this common nature, there is a critical lack of standard infrastructure to deal with movement.

While the collection of tracking data is growing on a daily basis, the use of R for movement studies has also increased sharply, in contrast to the decline of most other platforms—for instance, in the field of Movement Ecology, we counted more than 70 % of studies using R in 2018, i.e. roughly three times more than its closest contender, and the trend is still rising (Fig.1). The Movement community in R is at the same time very dynamic and very fragmented (Joo *et al.* 2018); we have listed 57 packages that process, visualize and analyze tracking data, one third of which worked in isolation, not being linked to any other tracking package. This is in part due to a lack of a modern infrastructure to deal with trajectories in R.

To date, existing classes in R for movement data are either outdated, relying on outdated standards, or too complicated to be used in a broad array of situations. Previous attempts notably include the following classes:

- `ltraj` (package `adehabitatLT`): Developed as early as 2006, this was the first class dedicated to movement that relied on a conceptual data model, based on the idea of successive steps, i.e. the straight-line segment connecting two successive locations (Calenge *et al.* 2009). However, the class itself is largely outdated, relying on an *ad-hoc* `list` structure, which is complicated to use, requiring to go back and forth between `ltraj` and `data.frame` classes frequently.
- `trip` (package `trip`): Also developed in 2006, `trip` is a S4 class directly extending `sp`'s `SpatialPointsDataFrame` class to connect series of points with a column of timestamps. The connection is made through a `TimeOrderedRecords` object that identifies which data columns corresponds to timestamps and identity of the moving object, making it a lightweight solution, however with limited flexibility for nested structures (e.g. split individual trajectories into different parts). Also, `sp` has been essentially superseded by the more recent package `sf` for spatial classes.
- `Move` (package `move`): Released in 2012, it is a complete S4 class focusing on animal movement data from the Movebank repository, but which can also be used on any tracking data. However, `Move` objects have a fairly complex structure, lack the flexibility of simpler classes, and are further arranged into

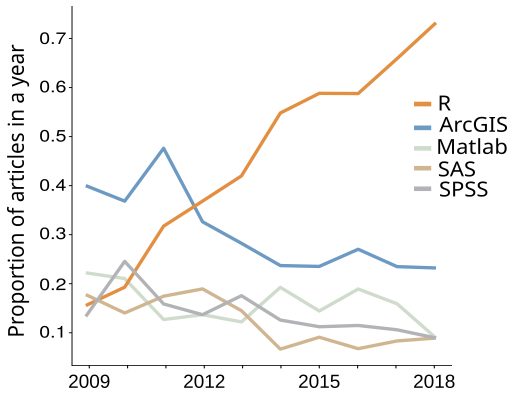

Figure 1: Main software used in Movement Ecology in the past ten years (2009–2018). Based on an extensive review of the literature, we listed 4417 papers of Movement Ecology, and extracted the software for analysis. Our review highlights the rise of R at the expense of all major contenders (Joo et al., in prep.).

`MoveBurst` (track split into different parts) and `MoveStack` (several tracks from different individual) classes. Moreover, `Move` objects also relies on spatial classes from `sp`.

- `Track` (package `trajectories`): Released in 2014, this is another example of formalizing a movement object as a S4 class, following roughly the same logic as the package `move`, with a complex nested class structure of `Track` objects (a single track from one individual), which are combined into `Tracks` (several tracks from the same individual) and `TracksCollection` (several individuals). `Track` also extends spatial classes from `sp`, and builds upon spatio-temporal classes from the `spacetime` package.
- `trackeRdata` (package `trackeR`): More recently released in 2016, this package focuses on running and cycling data from GPS-enabled tracking devices, and provide a specific class dedicated to athlete activity. Similarly to `ltraj` from `adehabitatLT`, `trackeRdata` are lists with locations stored as `zoo` objects (from the package of the same name), which are ordered, but not spatial, observations. This *ad-hoc* structure makes `trackeRdata` rather complicated to use.

Given the ubiquitous nature of tracking data, the fragmentation of the ecosystem of R packages dealing with such data, and the complexity of use of existing attempts at handling these data, it seems timely and relevant to address the issue from a data perspective.

The proposal

Overview

Based on years of experience, and broad feedback from the movement community, we aim to develop a central trajectory class to support all stages of movement studies (pre-processing, post-processing and analysis). We propose a `sftraj` package offering a generic and flexible approach . The only aim of the package will be to present this central class and basic functions to build, handle, summarize and plot movement data. Our project relies on three complementary pillars: a broad involvement of the movement community, a robust conceptual data model, and a `sf`-based implementation in R.

Detail

Definitions

Movement can be described by the (continuous) curve made by moving objects (Turchin 1998, Fig. 2). In practice, this curve is sampled (recorded) at discrete times to collect *locations*, which define *tracking data* of the form (x, y, z, t), i.e. vertices composed of 3-dimensional spatial coordinates plus the temporal coordinate. A *track* or *path* is thus composed of a series of locations, i.e. the "complete spatio-temporal record of a followed organism, from the beginning to the end of observations" (Turchin 1998, Biljecki *et al.* 2013).

Considering the temporal autocorrelation between successive locations as part of the movement information (Martin *et al.* 2009), we can then defines *steps* as the straight-line segments between two successive locations. Treating movement steps as straight lines is an "idealization" and the most parsimonious approach (other complex approaches exist in some fields, such as splines and Bezier curves), but at sufficient resolution, no significant information is lost. A sequence of steps finally forms a *trajectory*.

Movement is thus recorded by a series of locations in a track, and is abstracted as sequential steps in a trajectory. It is important to recognize this dual perspective between tracking data (locations) and movement data (steps) (Fig. 2).

Conceptual data model

To organize elements of data and how they relate to one another and to properties of the real world entities (here a moving object), we followed the methodology for conceptual modeling of geographic information that

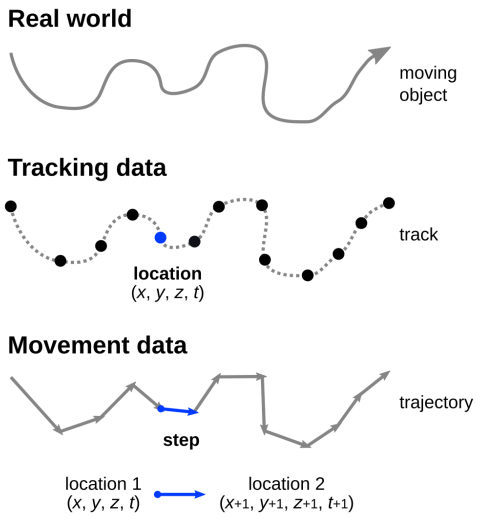

Figure 2: Definition of elements of movement.

is specified in the International Standards (ISO/TC 211) as described by Roswell (2011), which relies on a standard conceptual model specified in Unified Modeling Language (UML).

In the proposed model (Fig. 3), a `Location`, i.e. tracking data of the form (x, y, z, t) is implemented as an association between a `Point` (3D spatial coordinates) and a `Time` (temporal coordinate). A series (collection) of unique and ordered Locations then defines a `Track`. Two successive Locations also define a `Step`, as the elementary unit of movement data. A series (collection) of unique and ordered Steps finally defines a `Trajectory`.

Relationships between entities are not necessary symmetrical. It is only possible to navigate downwards from Track to Locations, and from Trajectory to Steps, as all information of a Location, or a Step, is contained in a Track, or a Trajectory, respectively, but not the other way around. Similarly, navigation is unidirectional between a Step and a Location, as two successive Locations are necessary to define a Step. However, navigation is bidirectional between a Track and a Trajectory, as it is necessary to enforce direct correspondence of both objects.

Implementation in R

The implementation aims at using existing building blocks as much as possible, as to not reinvent the wheel, and allow to use existing tools and functions on these building blocks. We thus propose to base our classes on simple `data.frame`s using `sf` spatial building blocks, structured as tidy data with observations (rows) and variables (columns). One of the variables is the track or trajectory column, which will store tracking data (tracks) and movement data (trajectories), respectively (Fig. 4). The location or step is the elementary observation and composes each row of the data frame. Ancillary data associated to the location or the step can be stored as additional variables.

The package `sf` provides the core spatial classes: Tracking data can be considered as `POINTS ZM`, i.e. points with longitude, latitude, altitude, and timestamps. If the exact time of the location is known, timestamps can be stored as `POSIXct`. Two successive `POINTS ZM` define a `LINESTRING ZM`, i.e. the straight-line segment between two successive locations. As the `M` coordinate has been little tested, its relevance and usefulness will be assessed early in the project; alternatively, timestamps will be stored separately from the geographic data. The `tsibble` package also presents an approach to deal with time series that will need to be further investigated.

Two new classes will be defined to store tracking data (`sftrack`) and movement data (`sftraj`), which

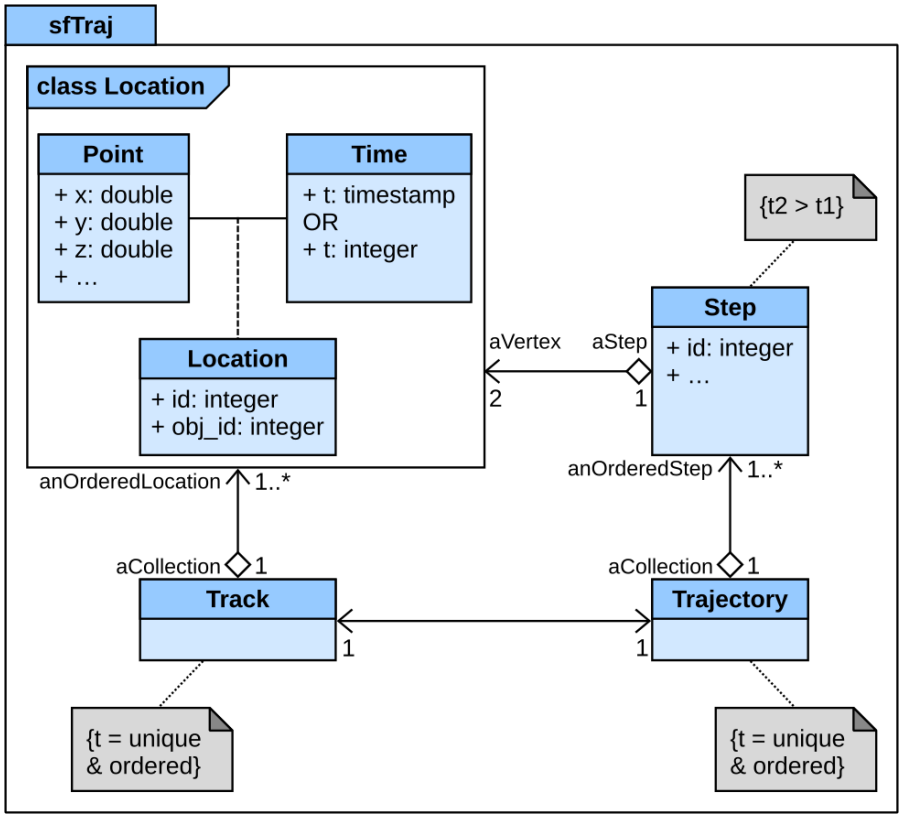

Figure 3: Proposed conceptual model for the 'sftraj' package using UML.

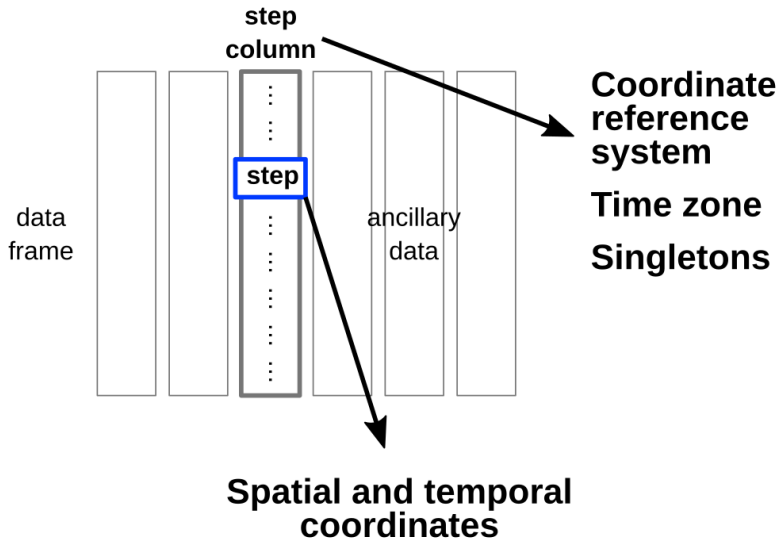

Figure 4: Proposed implementation of movement data in R. Tracking data are essentially similar, but do not have the 'Singletons' attribute.

both inherit from `sf` and `data.frame`. Correspondingly, the step list-column will be of class `sfc_TRACK` or `sfc_TRAJECTORY`, with a unique Coordinate Reference System (CRS) and timezone in the two (constant) attributes `crs` and `timezone`, respectively. The implementation is further complicated by the fact that a full correspondence between a *track* and a *trajectory* needs to be achieved, while there is no bidirectional navigation between a Location and a Step. An additional attributes `Singletons` specifically for `sftraj` will thus contain locations that do not form a step, i.e. in the case of a location surrounded by two empty locations (e.g. failed GPS attempts).

A single `sftrack` or `sftraj` object can be a single track or trajectory, or a set of them, sharing the same attributes (CRS and timezone). Nesting is an essential part of the data model, as it allows any relevant boundaries of a trajectory, such as the individual level, the year, or a journey (movement from home to work and from work to shopping, Biljecki *et al.* 2013), and any nested combination of those. The package `tidyr` provides a function `nest`, which allows to nest columns inside a data frame, and will be used to easily switch from one level to another (e.g. from an individual representation to an individual-year representation).

Project plan

Start-up phase

The `sftraj` package is meant to be a cornerstone for the development of a more cohesive Movement community in R. For this to happen, we will involve all voluntary contributors from the community, and work in the open all along. For instance, a collaborative platform already in place relies on a public GitHub repository. We will adopt the MIT license for the package, as to allow more contributions, and wide acceptation by other package developers.

The first stage of the work (see timeline below) will specifically involve the Movement community in R. During this stage, we will open contributions of *use cases* for the package (using GitHub's issue system), which set practical goals for the development of the package. Use cases describe the workflow that is expected from both users' and developers' perspectives, and thus the capabilities that a trajectory package needs to offer. The package specifications and development will aim at addressing all use cases described, to make sure the solution provided is relevant for a wide array of users and package developers.

Technical delivery

The time frame of this project is over 6 months. We used the MoSCoW method to determine what will be delivered from this project, starting from a minimum viable product to future development:

- **Must have** (requirements necessary for project completion, i.e. the minimum viable product):
  - Use cases described [month 1–2]
  - Data model revisited and classes definition [month 3]
  - Creators and converters from basic objects (`data.frames`, `sf`, `sftrack`, `sftraj`) [month 4]
  - Installable package (GitHub) [months 4–6]
  - Accessors and summaries (`print`, `summary`) [month 5]
  - Full function documentation and unit testing [months 5–6]
- **Should have** (important requirements not necessary for project completion):
  - Vignette [month 6]
- **Could have** (desirable requirements developed if time allows):
  - CRAN package
  - Basic plot (static)
- **Won't have** (requirements that are not planned at this stage):
  - See section on *Future work*.

Other aspects

We will publicly provide developments of new ideas and functions through regular blog posts on the MabLab website to communicate progress (basically for each stage and deliverable), and constant communication on Twitter by the core team. Twitter and emails will also be largely used to reach out to the community, as we have successfully done in the past for the R tracking package review (Joo *et al.* 2018) and the Movement Ecology synthesis (Joo *et al. in prep*).

Requirements

People

The core project team will initiate and lead the necessary consultation, code the proposed package, implement the data model, and ensure successful outcome of the project. All contributors to the proposal and people who have been consulted for it (listed above), and more broadly the entire Movement community, will also be consulted during the different phases of the work, and will be invited to contribute further to the realization of the project.

In particular, we request $10,000 in salary for 1.84 months of Matt Boone, Data Scientist at the University of Florida. Matt has a solid experience in R, and will be the main developer and contributor to the codebase. As all other contributions will be in-kind, having Matt fully dedicated guarantees successful completion of the project.

Processes

As this project ultimately aims at building a R Movement community, we will adopt a community-based code of conduct. The recently updated code of conduct of rOpenSci will serve as a basis, as we share foundational values of openness, tolerance and inclusiveness. We will adopt a "release early, release often" philosophy to provide most recent developments to the community, and allow for early testing of the `sftraj` package. At all stages but especially at the first one (*Use cases*), we will seek feedback from the Movement community. While working on the review of R tracking packages (Joo *et al.* 2018), we have been successful in mobilizing the Movement community through Twitter and emails notably; for instance, 225 persons filled out our survey on package usage and documentation. We will use this already established connection to further reach out, and invite the largest number of people to contribute and provide feedback. Public development on GitHub will also allow the use of the issue system to initiate and discuss each use case with and from the community.

Tools & Tech

No technical constraint is foreseen. All team members are already equipped with enough computer power to work on the project. The development platform (GitHub) is already set up and public, and will remain open to the entire Movement community.

Funding

We request a total of $10,000 to support 1.84 months of Matt Boone, Data Scientist (Biological Scientist II) in the MabLab at the University of Florida.

- Salary: $7,369
- Fringe rate (35.70 %): $2,631
- **Total award = $10,000**

Summary

Salary to support a Data Scientist is requested to have one person committed to the project, who will dedicate set chunks of time to the work. This seems required to ensure project completion in the proposed timeline. Almost 8 weeks of work is a reasonable amount of time, which matches proposed deliverables. All other contributions, from core team members and other contributors, will be in-kind.

Success

Definition of done

This project will be successful if a functional R package (`sftraj`) that provides usable classes for tracking and movement data is available at the end of the 6 months period. This will pave the way for further consolidation and adoption by the Movement community in the next stages of the `sftraj` project.

Measuring success

Success can be measured by the completion of deliverables through time, as set up in the *Project plan* section (see notably the minimum viable product). Beyond deliverables, success can also be measured both during and after the period of work by the adoption by the Movement community:

- **During the work:**
  - Contributions of the Movement community during the early stage (*use cases*): number of contributors and contributions;
  - Beta testing from the Movement community: number of testers and issues submitted;
  - Code contribution: number of contributors and contributions.
- **After the work:**
  - From a user perspective: number of downloads (e.g. using RStudio download statistics);
  - From a developer perspective: adoption in tracking packages.

Future work

The long-term plan for `sftraj` includes major steps relying on the availability of a functional trajectory package. Future stages of development already include:

- Full-fledged package: submission to CRAN and to rOpenSci.
- Preparation of a detailed article (targeting the R journal) to present the technical choices and the solution offered by `sftraj`, in order to favor adoption by users and package developers.
- Keep the conversation open with package developers, and help them develop conversion tools to major existing classes.
- Dynamic visualization of trajectories, allowing keyboard- and mouse-controlled exploration of trajectories, step by step (based on the solution provided in `rpostgisLT`).

Other avenues of development to be explored further include:

- Supporting measurement units and errors, and their propagation, in the spatial components of the trajectories, i.e. making the package `units` and `errors` compliant.
- Developing tools to clean and interpolate trajectories, based on specific filters and assumptions.

Key risks

Delays on the project could happen because of community feedback that requires reformating classes or halting it altogether because of a lack of consensus. This can be mitigated by making sure the Movement community is largely involved and consulted at every stage of the project. This is a fundamental aspect of our approach.

Technically, the main hurdles may deal with the structure of `sf` objects, which, if not resolved directly, may require the consultation and/or collaboration of Edzer Pebesma, author of the `sf` package, who contributed to the development of this proposal.

Finally, there are inherent risks related to unforeseen problems that will arise and the extra time necessary to address them. This project has been framed with discrete manageable goals for every month of the period of work. We defined reasonable deliverables with the MoSCoW method, including optional and desirable requirements if time allows. We also made sure to keep unrealistic requirements out of the scope of this grant (see *Future work* above).

References

- Biljecki, F., Ledoux, H., & van Oosterom, P. (2013). Transportation mode-based segmentation and classification of movement trajectories. International Journal of Geographical Information Science, 27, 385–407. https://dx.doi.org/10.1080/13658816.2012.692791
- Calenge, C., Dray, S., & Royer-Carenzi, M. (2009). The concept of animals' trajectories from a data analysis perspective. Ecological Informatics, 4, 34–41. https://dx.doi.org/10.1016/j.ecoinf.2008.10.002
- Gudmundsson, J., Laube, P., & Wolle, T. (2011). Computational Movement Analysis. *In* Kresse W, & Danko D. M. (Eds.), Springer Handbook of Geographic Information (pp. 423–438), Springer-Verlag Berlin Heidelberg. https://dx.doi.org/10.1007/978-3-540-72680-7_22
- Joo, R., Boone, M. E., Clay, T. A., Patrick, S. C., Clusella-Trullas, S., & Basille, M. (2018). Navigating through the R packages for movement. arXiv, 1901.05935, https://arxiv.org/abs/1901.05935
- Martin, J., Tolon, V., Van Moorter, B., Basille, M., & Calenge, C. (2009). On the use of telemetry in habitat selection studies. *In* D. Barculo, & Daniels J. (Eds.), Telemetry: Research, Technology and Applications (pp. 37–55), Nova Science Publishers Inc.
- Roswell, C. (2011). Modeling of geographic information. *In* Kresse W, & Danko D. M. (Eds.), Springer Handbook of Geographic Information (pp. 3–6), Springer-Verlag Berlin Heidelberg. https://dx.doi.org/10.1007/978-3-540-72680-7_1
- Turchin, P. (1998). Quantitative analysis of movement: measuring and modeling population redistribution in animals and plants. Sinauer Associates.