Australian Marine Mammal Centre

Final Report

(subclause 9 and Schedule Item 5 of the Funding Agreement)

- **Project No**. – 2009/39
- **Title** - Track: A new tool for integration, interpolation and visualisation of animal tracking data
- **Chief Investigator** – Professor Mark Hindell
- **Organisation** – University of Tasmania
- **Report Author and Project Developer** – Sascha Frydman

Activity Period – **January 2010 – May 2011**

Table of contents

1. Activity Summary

A clear summary of approximately 500 words outlining the work undertaken and any significant findings (for publication on the Department's web site)

Marine mammal movement data is primarily available from animal-borne telemetry systems as infrequent, unpredictable locations of various accuracy on the Earth's surface. Additional rich and precise information about depth in the water column can be obtained with animal-borne data loggers. Determining where an animal might go in the periods between known locations is called interpolation and requires sophisticated, mathematically based computational techniques and software. This project delivers a software application called Track that will enable researchers to more easily and accurately predict marine mammal movements. By building on the well established R analysis platform, this system is able to leverage existing functionality while adding facilities for the storage and analysis of animal movement data. The initial version of this system is focussed on data supplied by the Argos satellite system. However, it is envisaged that later releases will provide for input from other geographical data sources such as FastLoc GPS.

The utility of bio-logged and remote sensed data is dependant on a quantitative, accurate and defendable extraction and interpretation of the underlying biological signal. Point location data are usually biased and can lead to erroneous interpretations of where animals actually spend most of their time. Only through more sophisticated interpolation models can more realistic estimates of animal space utilization patterns be determined. A key output of the Track system is a map showing the likelihood of finding an animal or animals at a given location. This map can therefore be used to estimate potential habitat usage. The quality of this estimation is determined by the quantity of data supplied to the system, ie. as more data is collected, knowledge of habitat usage will become more accurate and precise.

An integral part of the design of this system is a readily extendible framework whereby users can implement new methods for manipulating the interpolation algorithm. For example, the system currently uses information about animal travel speed and diving behaviour to refine the estimate of spatial area usage but these constraints can be easily modified or replaced by other calculations. By conforming to existing standards for spatial data representation, these results are then readily interpreted and visualised both within the Track package as well as through the use of other freely available R spatial analysis packages.

Development of this system took place in conjunction with a PhD project titled, ‘Analytical Techniques for the Interpretation of Satellite Derived Marine Animal Locations’. The thesis for this PhD contains extensive quantitative assessment of the application of the Track system. A significant development described in this work is the algorithm called Model Interpolated Kernel Smoothing. This algorithm which is one of the core functions of the Track system was found to outperform other techniques such as location time weighting and linear interpolation when used in the creation of spatial usage maps.

2. The Outcomes/Objectives

List of the Project Objectives

The objectives as listed in the original proposal are:

1. Database facility for storage and management of location and dive data and associated metadata
2. Visualisation and mapping tools that will enable simultaneous exploration of any number of locations, tracks and dives in 2D, 3D and with animations through time
3. Analysis system for filtering, interpolation and integration based on algorithms developed by Sascha Frydman
4. Ability to incorporate user defined algorithms to expand the functionality of the system beyond its current scope
5. Links between the systems core algorithms and existing analysis environments including R and Matlab
6. Additional requirement for a workshop on the use of this system

The degree to which the Activity has achieved each of the objectives

1. A relational database system for storing the raw location data has been implemented using the SQLite database engine. A set of R functions for the management and extraction of data for analysis and visualisation are also provided.
2. Visualisation and mapping is achieved using standard R techniques. Integration with currently available mapping and spatial analysis packages is achieved through the use of the sp package. The sp package provides a framework for implementing spatial data types as well as generic map plotting functions. The Track package contains some ‘canned’ examples of how to visualise its data. These are written in plain R language format so as to be easily modified for other applications.
3. The Track system was used as the primary analytical tool in the production of a

recently submitted PhD thesis. This thesis is titled, ‘Analytical Techniques for the Intepretation of Satellite Derived Marine Animal Locations’. Extensive documentation on the theory underlying the interpolation algorithm as well as the design and structure of the software can be found in the body of this thesis. This thesis also contains quantitative assessments of the performance of the interpolation algorithm and the application of kernel density estimation in the generation of spatial utilisation distributions. Since the thesis is currently with the examiners, it is not yet available for public release. I have however attached a copy with this report as proof of the application and validation of the Track system to real world data. Once the thesis has been accepted, users of the system will be able to download its chapters from the University of Tasmania website. In addition to this I intend to partially rewrite and in places extend these chapters in preparation for publication in appropriate journals. These papers will then become the primary source of information on the theory and design of the system as well as the mechanism for citation when referring to the system’s use in subsequent studies.

4. Expansion of the system’s core algorithms is achieved through the development of object oriented (OO) sub-classes of predefined ‘Objective’ and ‘Observer’ base classes. The latest technology for OO programming in R, known as reference classes is used. This system provides a more traditional OO approach and is far simpler to understand and use than the alternative, S4 methodology. While it is still necessary for an ‘expert’ user to fully understand the theory behind the optimisation algorithm, once designed, it is quite straight-forward to actually implement these additions.
5. The objective for linking to other analysis environments is no longer valid since the Track system is now housed entirely within R.
6. Delivery of a workshop on the use of this system is expected to take place in the next couple of months. The timing of this will be made to coincide with my return to Hobart for the formal completion and final submission of my thesis. My plan for this workshop will be a full day with the first half devoted to the theory of the system and the second half of the day used to demonstrate its application. There is also the possibility of extending this workshop into two days by spending a day teaching techniques on developing high performance code in R using C and C++. While not vital to understanding the use of Track, this additional day would provide a valuable precursor to exploring the Track system, especially if the user plans on modifying or extending its functionality.

Following is a list of functions and classes provided by the Track package.

`db.create`	Creates a new, empty instance of the Track database with all tables, indexes and key constraints in place.
`db.delete`	Delete specified data from database.
`db.store.argos`	Read and parse Argos location data and store in connected database.
`db.store.tdr`	Store time depth recorder data in database.
`db.metadata`	Add or modify track metadata.
`db.summary`	Print statistics on the current data stored in the Track database.
`db.tracks`	Retrieve Argos location data grouped by

	PTT as defined in track metadata. Returns sets of locations grouped as one or more individual animal movement tracks.
TOptimiser	Track optimiser reference class. This is the core algorithm that uses extremal optimisation to explore the uncertainty surrounding Argos location data. This is the framework that the various objective, observer and mutation objects are plugged into. See the speed.filter.optimising and location.interpolation functions for working examples of its application.
TMutator	Base class for implementing location mutation operators.
TObserver	Base class for developing optimiser observers.
TObjective	Base class for developing optimiser objectives.
ArgosLocationMutator	Random mutation operator that adjusts the coordinates of Argos locations within the confines of the estimated observation error.
InterLocationMutator	Random mutation operator that adjusts the coordinates of the interpolated location through the use of an adaptive equation that is a function of the position and time of adjacent Argos locations.
ArgosErrorObjective	Objective that constrains the optimisation to regions defined by the estimated location error surrounding each position fix.
MaxSpeedObjective	Objective that produces track solutions that do not exceed a given maximum speed.
make.speedPDF	Produces a probability density function of speed based on a supplied set of multiple location tracks.
SpeedPDFObjective	Objective that uses goodness-of-fit to tune the optimisation interpolations so as to closely match a predetermined population level probability distribution function.
TDRObjective	Objective that uses TDR dive data to create an index of vertical displacement that is in turn used to limit the optimisation system's level of horizontal displacement.
SelectionObjective	Used by optimiser to uniformly distribute iterative location selection.
LocationRecorder	Simple observer that monitors and records the sequential changes made to the track locations as a consequence of running the optimiser.
GridPointRecorder	Optimiser observer that provides equi-temporal interpolated track location sampling.
deg2rad	Conversion from degrees to radians.
gcd.hf	Calculates the geodesic distance between two points specified by radian latitude, longitude using the Haversine formula.
gwkde.SpatialGrid	Geographical coordinate data based weighted kernel density estimation. This

	function takes decimal degree location data with or without a weighting attribute and produces an unprojected SpatialGrid. Provided are several options for determination of smoothing parameter. These include manual selection, href method, least squares cross-validation and likelihood cross-validation.
speed.filter.mcconnell	Implementation of the iterative RMS speed filter described in McConnell et. al. (1992)
speed.filter.optimising	Speed filter that utilises the new optimising system to incorporate knowledge of Argos location error into estimates of an animal's spatial movements.
location.interpolation	The main processing function for the system. This function utilises most of the classes and functions contained in this package to produce a set of interpolated equi-temporally distributed pseudo-locations.
ud.location	Generates an 'sp' based SpatialGrid that represents an estimate of the animal's utilisation distribution. The input to this function is a track or set of tracks of geographical coordinate locations.
ud.linear	This function linearly interpolates track locations prior to generating the utilisation distribution.
ud.MIKS	Model interpolated kernel smoothing. Produces a utilisation distribution using the output of the location.interpolation function.
plot-methods	Convenience method that brings together plotting functions supplied by sp and lattice to generate pre-defined visualisations of the Track system's datasets and interpolation results.

3. Appropriateness

The appropriateness of the approaches used in the development and implementation of the Activity

The original specification for the Track system detailed the use of the C++ based visualisation toolkit (VTK) in conjunction with the data visualisation platform, ParaView. During the course of last year, this system was developed and much of the functionality made operational. However, extensive use of ParaView highlighted a flaw in this approach. When incorporating external components, it was found to be somewhat 'buggy' and not as robust or easy to use as early preliminary testing had indicated. Since the intention was always to supply some functionality within the R statistical analysis system, it became clear that the solution would be to shift the focus to a more complete R implementation.

One advantage with using R is that many animal tracking researchers use this system and as such, several packages with complementary functionality already exist.

That being said, a pure R implementation of the proposed optimisation algorithm is not feasible due to the interpreted nature of the language and its relatively slow computational performance. Given that much of the functionality for the Track system had already been implemented in C++, the decision was made to migrate the core parts of this code across to an encompassing R package. A number of R packages that have only recently become available have aided in this migration. In particular, the introduction of reference classes into the core R library as well as the ongoing development of the now quite sophisticated Rcpp package. The Track system as it now stands, employs C and C++ to implement the more computationally intensive functions, e.g. calculation of great circle distances and most of the core optimisation algorithm. R provides the interface that the user interacts with as well as being the 'glue' that connects all the components together, thereby establishing the dataflow from raw input data through to processed output.

The R version of the system manages all data import, export, manipulation and visualisation. Where appropriate, high speed code is written in C or C++ and works directly on the native R data types. As is common practice in many R packages, the existence of external compiled code is noted in the documentation but beyond that does not impact the functionality or user experience when compared to functions written entirely in R. A number of currently available packages provide core functionality within the Track system. Their use offers a level of standardisation that permits the output of the Track system to be easily integrated into workflows created with other packages. These packages are as follows,

- **sp** - provides classes and methods for handling and visualising spatial data
- **lattice** - a high-level data visualization system with an emphasis on multivariate data
- **methods** - formally defined methods and classes for R objects, in particular, the new implementation of reference classes that provide for a more typical style of object oriented programming in conjunction with a true pass-by-reference protocol
- **Rcpp** - provides a C++ library which facilitates the integration of R and C++.
- **inline** - functionality to dynamically define R functions and S4 methods with in-lined C, C++ or Fortran code
- **RSQLite** - database interface R driver for SQLite. This package embeds the SQLite database engine in R and provides an interface compliant with the DBI package

4. Effectiveness

The degree to which the Activity has effectively met its stated objectives

The utility and effectiveness of the Track system in the analysis of real world animal tracking data is demonstrated by its use throughout the attached PhD thesis. All results were produced using data stored within the Track database and processed and analysed with the algorithms and functions provided by this system. The system provides a robust self-contained database facility for storing and managing marine animal movement data. Furthermore it introduces and makes available new techniques for interpolation and spatial usage estimation of these data.

The decision to recreate the Track software within the R environment has unavoidably introduced additional technical and development time challenges over that provided by the original project proposal. Through the use of the various R packages described above I have managed to successfully integrate the original C++ system into the new environment. However, the final polished R package is currently being worked on and will be ready for wider use by the end of July 2011.

Official public release of the system will be timed to coincide with the availability of the thesis chapters and prior to the workshop taking place. The system will be distributed as an R package and will therefore be made available through the Comprehensive R Archive Network (CRAN).

5. Communication

How results will be communicated to management

The outcomes of this work will be made available via free download across the internet using the package distribution network provided by the R statistical analysis environment.

Communication of this work will be achieved through the planned user workshop, availability of a PhD thesis which makes extensive use of the system and peer reviewed publications.