

Maximum likelihood estimation for randomized shortest paths with trajectory data

ILKKA KIVIMÄKI[†]

*Department of Computer Science, Aalto University, Espoo, Finland and
Université Catholique de Louvain, ICTEAM, Louvain-la-Neuve, Belgium*

[†]Corresponding author. Email: ilkka.s.kivimaki@gmail.com

BRAM VAN MOORTER, MANUELA PANZACCHI

Norwegian Institute for Nature Research, Trondheim, Norway

JARI SARAMÄKI

Department of Computer Science, Aalto University, Espoo, Finland

AND

MARCO SAERENS

Université Catholique de Louvain, ICTEAM, Louvain-la-Neuve, Belgium

Edited by: Ernesto Estrada

[Received on 18 August 2018; editorial decision on 1 July 2020; accepted on 12 July 2020]

Randomized shortest paths (RSPs) are tool developed in recent years for different graph and network analysis applications, such as modelling movement or flow in networks. In essence, the RSP framework considers the temperature-dependent Gibbs–Boltzmann distribution over paths in the network. At low temperatures, the distribution focuses solely on the shortest or least-cost paths, while with increasing temperature, the distribution spreads over random walks on the network. Many relevant quantities can be computed conveniently from this distribution, and these often generalize traditional network measures in a sensible way. However, when modelling real phenomena with RSPs, one needs a principled way of estimating the parameters from data. In this work, we develop methods for computing the maximum likelihood estimate of the model parameters, with focus on the temperature parameter, when modelling phenomena based on movement, flow or spreading processes. We test the validity of the derived methods with trajectories generated on artificial networks as well as with real data on the movement of wild reindeer in a geographic landscape, used for estimating the degree of randomness in the movement of the animals. These examples demonstrate the attractiveness of the RSP framework as a generic model to be used in diverse applications.

Keywords: randomized shortest paths; random walk; shortest path; parameter estimation; maximum likelihood; animal movement modelling.

1. Introduction

1.1 Background and motivation

Modelling the movement and flow of different entities on networks is a key topic in network science [1–6]. It is not only relevant for studying physical movement but also for more abstract processes

such as communication, the spread of diseases and financial transactions, to name a few. Moreover, models of movement and flow are often used as the basis of popular distance, centrality and other measures on networks.

The two most standard paradigms for studying movement on networks consider it as occurring over shortest paths or over random walks. These paradigms are, however, often too simple for building realistic models. In reality, movement or flow rarely strictly follows the shortest paths, nor is it completely random. In addition, the standard network measures derived from these paradigms often have caveats. The shortest path distance, for instance, does not take into account connectivity besides the shortest path, that is, the number of other short connections between nodes. Moreover, when comparing distances on unweighted graphs, the shortest path distance often results in ties between node pairs, because it typically yields a limited number of integer values.

Random walks can be used for defining the *commute time* and *commute cost distances* [3, 7, 8], as well as the closely related *resistance distance* [9]. They reflect the connectivity between nodes, but only in small networks. Instead, in large networks they become only dependent on the local connectivity of the nodes, in many cases the node degrees [10, 11]. This phenomenon has been called the *global information loss problem* in [12, 13].

The above reasons have motivated the development of alternatives for the traditional paradigms, with special focus on defining new distances on networks. One such alternative, which is also the main topic of this work, is the *randomized shortest paths (RSPs) framework* [8, 14–16]. The RSP framework is based on considering a Gibbs–Boltzmann distribution over paths from a *source node* s to a *target node* t involving an *inverse temperature parameter*, $\beta = 1/T > 0$, which controls the degree of randomization from the optimal, shortest paths. At the limits of the parameter range, the distribution focuses either on solely the optimal paths ($\beta \rightarrow \infty$) or spreads over random walks ($\beta \rightarrow 0^+$). In the framework, the optimality of a path can be defined as the number of steps, or, more generally, based on real-valued *edge costs*, denoted by c_{ij} for edge (i, j) . On the other hand, random walks are considered based on *edge affinities* a_{ij} which can be defined independently of the edge costs (see Section 1.2 for details). This enables the consideration of optimality and randomness of paths as being based on different grounds, which can be relevant for some applications. The Gibbs–Boltzmann distribution is an exponential-family distribution, but can also be interpreted in terms of maximum entropy modelling [17].

Originally the RSP framework was proposed in order to define many interpretable metrics on a network, such as distance measures between the network nodes [8, 16] or node and edge centrality measures [14]. Furthermore, the RSP framework can also be used for purposes of planning or predicting paths on networks and for modelling movement or flow patterns on networks [18, 19], which is the application domain considered in this article. One problem that has not yet been tackled in depth in the literature is the estimation of the parameters of the RSP model. This work tackles the problem by presenting methods for computing maximum likelihood estimates (MLEs) of the parameters of the RSP model in cases where the data consist of trajectories between nodes on a network. Such data may be generated whenever recording a movement or flow process on a network, including examples such as

- geolocation data of animals moving on a geographical network,
- trip data gathered from public transportation networks,
- trajectory data of players in video games, or
- browsing behaviour data of web users.

The main focus in this work is on the estimation of the inverse temperature parameter β based on such data, in which case the network structure, that is, the edge costs and edge affinities, are considered known and fixed. A relatively high value of β (i.e. low temperature) describes trajectories following optimal or near-optimal paths between the source and target nodes, whereas a low value of β (high temperature) can describe trajectories resembling a random walk with a drift towards the target [18]. By setting β to an appropriate value, the RSP model can take into account the assumption that movement or flow on a network often does not occur completely optimally nor completely randomly. When considering movement on a network, such an assumption can be valid for various reasons. For example, the agent (e.g. an animal) moving on the network might not have sufficient global information of the network or be intelligent enough in order to follow optimal paths. Or the agent can simply have a simultaneous preference for both randomness (i.e. exploration) and optimality (exploitation), which can help in obtaining more knowledge of the environment or in distracting a prey or an opponent trying to predict and intercept the agent.

In earlier literature, the value of β has often been selected by some form of tuning. However, these tuning schemes are not based on solid statistical grounds and can be very specific for the application in question. For instance, in [8], the authors use distance measures derived from the RSP framework for clustering graph nodes, and the value of β is fixed by maximizing the clustering performance using a part of the data.

We first derive the MLEs of β for trajectories that have been recorded fully but also tackle the more complicated scenario where the trajectories are incomplete. We confirm the validity of these methods with artificial examples using simulated geographic networks and an artificial community-structured network. In addition, we test the developed MLE methods on real data of trajectories in a geographic landscape, based on GPS recordings of movements of wild reindeer. This example is devised mostly to verify that the MLE computations can be run in practice and that the methods provide reasonable results. Recently, the RSP framework was applied to similar data in [19] for locating movement corridors of wild reindeer in a geographical landscape. In addition, it has been made familiar to the ecological community thanks to the implementation in the `gdistance` R package [20]. This article is partly a continuation of the work in [19], as we also consider trajectories of wild reindeer as a use case for the problem of fitting the model to data of movement trajectories. However, in addition to this ecological application, the theory developed in the article is generic so that the methods can be used for any movement or flow phenomena on networks that are suitable for the RSP model.

In addition to β , the edge costs, c_{ij} , and edge affinities, a_{ij} , can be considered as parameters of the RSP framework. Although we do also briefly discuss the estimation of edge costs from trajectory data, in Section 2.4, the focus is mostly on the estimation of β , as commonly in RSP applications the graph structure, that is, the edge costs and affinities, are considered fixed and known. Furthermore, as the purpose of this work is only in developing the RSP theory for fitting the RSP model to trajectory data, the evaluation of the MLEs in specific ecological applications and comparison with other model estimation approaches are left for future work. We also plan to later investigate the use of the methods developed here for parameter estimation in the context of clustering and classification of graph nodes.

The article structure is as follows: in the rest of this section, we recall the RSP framework and discuss its properties that are required for following the remainder of the article. We also discuss other work related to model estimation in similar problem settings at the end of the section. In Section 2, we derive and validate MLEs for data consisting of complete trajectories. Section 3 deals with MLEs in situations where the data consist of incomplete trajectories, that is, where only a subset of the edge or node sequence of a path is observed. In Section 4, we use the methods derived in Section 3 to fit the RSP model to data

consisting of GPS trajectories of wild reindeer by estimating the inverse temperature parameter associated to the trajectories.

1.2 Randomized shortest paths

The RSP framework has been developed during the past decade in several works [8, 14–16], focusing mostly on the definition of distance, similarity or centrality measures on graphs or networks. It was initially inspired by models developed in the field of transportation science [21]. In this section, we recall the definition of the RSP framework and the related results relevant for the remainder of the article. Before that, however, we define the terms and notation used in the article.

1.2.1 Definitions and notation

Vector and matrix notation. Vectors are generally denoted by lowercase boldface characters and matrices by capital boldface characters, \mathbf{e}_i denoting the i th basis vector, that is, 1 at element i and 0 elsewhere. The length of vectors and size of matrices is determined depending on the context, if not stated explicitly. \mathbf{I} denotes the identity matrix (of appropriate size) and \mathbf{I}_{ij} denotes the matrix whose element (i, j) is 1 and other elements are 0. Otherwise, for an arbitrary matrix \mathbf{X} , the lowercase x_{ij} and sometimes $[\mathbf{X}]_{ij}$ are used to denote the element (i, j) of \mathbf{X} , whereas \mathbf{X}_{ij} denotes the matrix containing the value $[\mathbf{X}]_{ij}$ at element (i, j) and zero elsewhere.

Graphs and paths. Let $G = (V, E)$ be a directed, strongly connected graph with node set V containing n nodes labelled from 1 to n , i.e. $V = \{1, \dots, n\}$; and edge set E containing m edges represented as ordered pairs (i, j) , where $i, j \in V$ and $i \neq j$ (i.e. we do not consider graphs with edges from a node to itself). For any node $i \in V$, we denote by $Succ(i)$ the set of *successor* nodes of i , that is, $Succ(i) = \{j \in V \mid (i, j) \in E\}$.

A *path* \wp on G is defined as a sequence of nodes $\wp = (v_0, \dots, v_L)$, where $(v_{l-1}, v_l) \in E$ for all $l = 1, \dots, L$ and where $L \geq 1$ is the *length* of the path (note that thus a single node does not constitute a path). The length of an arbitrary path \wp is denoted as $L(\wp)$. The l th node of path \wp is denoted by $\wp(l)$, where $0 \leq l \leq L(\wp)$. Note that the node indexing along a path starts from 0. Likewise, the edge between the l th and $(l+1)$ th node is denoted by $\wp(l, l+1)$, and the *subpath* or *subsequence*, from the l_1 th to the l_2 th node by $\wp(l_1 : l_2)$.

The focus in this work is especially on *hitting paths*, that is, paths, where the last node appears only once, or, formally, paths \wp for which $\wp(i) \neq \wp(L(\wp))$ for all $i < L(\wp)$. The set of all hitting paths from a starting node s to a target node t is denoted by \mathcal{P}_{st} . Note that with the above definition, a hitting path cannot go from a node to itself, and thus $\mathcal{P}_{tt} = \emptyset$ for all $t \in V$. Moreover, the subset of \mathcal{P}_{st} containing only paths of a fixed length k is denoted by $\mathcal{P}_{st}^{(k)}$. A pair of starting node s and target node t is concisely referred to as an s - t -pair and the paths in \mathcal{P}_{st} as s - t -paths. Throughout the article, t is used as the index of the *absorbing* target node. Note that the results derived in this work for hitting paths can be easily generalized to all paths, not only hitting ones. We focus only on hitting paths for two reasons: because of brevity, and because the network measures derived from the RSP framework are more directly related to many traditional network measures when considering hitting paths than when considering all paths.

Edge weights, path costs and probabilities. In the RSP framework, each edge $(i, j) \in E$ is associated with two kinds of weights: an affinity $a_{ij} > 0$ and a cost $c_{ij} > 0$. In addition, for node pairs that are not connected by an edge, that is, such $(i, j) \in V \times V$ that $(i, j) \notin E$, we define $a_{ij} = 0$ and $c_{ij} = \infty$. The edge affinities and costs define the *affinity matrix* \mathbf{A} and *cost matrix* \mathbf{C} , both of size $n \times n$, whose elements (i, j) are the corresponding values for the node pair $(i, j) \in V \times V$.

The edge costs c_{ij} define the optimality of movement and can be considered in terms of the energy consumption, geographical distance, time duration, monetary value or other form of expense related to the step over an edge. The cost of a path \wp is defined as the sum of the edge costs along the path:

$$\tilde{c}(\wp) = \sum_{l=1}^{L(\wp)} c_{\wp(l-1),\wp(l)}. \quad (1.1)$$

Note the tilde above the c , which we use generally to differentiate between path-related and edge-related quantities. For any s - t -pair, the *least cost* from s to t means the minimum path cost over all s - t -paths. The least cost can be considered as a directed and weighted form of the *shortest path distance*.

While the edge costs define optimality of movement, the edge affinities instead determine what is meant by random movement. Namely, the affinities determine random walks on the graph, which are generated by the transition probability matrix \mathbf{P}^{rw} with elements

$$p_{ij}^{\text{rw}} = \frac{a_{ij}}{\sum_{k=1}^n a_{ik}}, \text{ for all } (i,j) \in E, \text{ and } 0 \text{ otherwise.} \quad (1.2)$$

The random walk with the above transition probabilities corresponds to a first-order Markov chain with state set V . The *random walk probability distribution*, $\mathbf{P}_{st}^{\text{rw}}$, over the set of hitting paths from s to t is determined by the product of the transition probabilities, that is, for any hitting path $\wp \in \mathcal{P}_{st}$,

$$\mathbf{P}_{st}^{\text{rw}}(\wp) = \prod_{l=1}^{L(\wp)} p_{\wp(l-1),\wp(l)}^{\text{rw}}. \quad (1.3)$$

Note that this product determines the path probabilities in a well-defined manner, as it is known that for hitting paths (see e.g. [22])

$$\sum_{\wp \in \mathcal{P}_{st}} \mathbf{P}_{st}^{\text{rw}}(\wp) = 1. \quad (1.4)$$

The random walk generated by the transition probabilities p_{ij}^{rw} is sometimes referred to as the *reference*, *natural*, or *unbiased* random walk and the superscript ‘rw’ in the above quantities naturally refers to (unbiased) random walks.

The reference random walk defines the *expected hitting time* and the *expected hitting cost* from s to t , as the expected path length, or path cost, respectively, over the random walk distribution $\mathbf{P}_{st}^{\text{rw}}$. The sums, from s to t and back from t to s , of expected hitting times and costs define, respectively, the *commute time distance* and the *commute cost distance*. It is well known that on an undirected graph (i.e. a graph for which $a_{ij} = a_{ji}$ and $c_{ij} = c_{ji}$ for all $(i,j) \in E$), where edge costs correspond to edge *resistances* (i.e. $c_{ij} = r_{ij} = 1/a_{ij}$; in which case affinities correspond to conductances), both the commute time and commute cost distances are proportional to the *resistance distances* between corresponding nodes. More exactly, if we denote, for any s - t -pair on an undirected graph, by Δ_{st}^{res} , Δ_{st}^{CT} and Δ_{st}^{CC} , respectively, the resistance, commute time and commute cost distances between s and t , then the following holds [7, 8, 23]:

$$\Delta_{st}^{\text{res}} = \frac{\Delta_{st}^{\text{CT}}}{\sum_{(i,j) \in E} a_{ij}} = \frac{\Delta_{st}^{\text{CC}}}{\sum_{(i,j) \in E} a_{ij} c_{ij}}. \quad (1.5)$$

As mentioned earlier, in Section 1.1, although the edge costs can be defined based on the edge affinities (for instance as $c_{ij} = 1/a_{ij}$, as above in the electric circuit analogue), generally in the RSP framework, the two weights can be—and are considered in the derivations as— independent of each other. This assumption is convenient technically but also makes sense due to the fact that the underlining drivers behind optimal and random movement can be very different from each other. In the simplest examples, even when the edge costs may obtain arbitrary non-negative values, the random walk can be considered oblivious to the edge features by defining the affinities $a_{ij} = 1$ for all $(i, j) \in E$, which results in a natural random walk with uniform transition probabilities at each node. On the other hand, defining $c_{ij} = 1$ for all $(i, j) \in E$ corresponds to considering the length of paths as their cost, in which case the random walk can nevertheless be considered with arbitrary affinities that depend on local features of edges, thus leading to a natural random walk with non-uniform transition probabilities at each node.

1.2.2 Definition of the RSP framework The RSP framework can be defined and interpreted in different ways [8, 14–16, 22], but here, we present the most often formulated version, based on *cost minimization constrained by relative entropy*. Consider an agent moving on the graph from a starting node s to a target node $t \neq s$. Instead of moving randomly, as defined by the unbiased transition probabilities in Equation (1.2), the agent aims to move in an optimal way. However, for various possible reasons, the movement of the agent is not completely optimal.

Instead, the agent is considered to choose its path from \mathcal{P}_{st} from a distribution which minimizes the expected path cost constrained to a fixed relative entropy with respect to the unbiased random walk. Formally, we seek for the distribution that satisfies the minimization problem

$$\text{Minimize}_{\mathbf{P}_{st}} \langle \tilde{c} \rangle = \sum_{\wp \in \mathcal{P}_{st}} \mathbf{P}_{st}(\wp) \tilde{c}(\wp) \quad \text{s.t.} \quad \begin{cases} \mathbb{J}(\mathbf{P}_{st} \parallel \mathbf{P}_{st}^{\text{rw}}) = J_0 \\ \sum_{\wp \in \mathcal{P}_{st}} \mathbf{P}_{st}(\wp) = 1, \end{cases} \quad (1.6)$$

where $\mathbb{J}(\mathbf{P}_{st} \parallel \mathbf{P}_{st}^{\text{rw}})$ is the *Kullback–Leibler divergence*, or *relative entropy*, with respect to the natural random walk distribution

$$\mathbb{J}(\mathbf{P}_{st} \parallel \mathbf{P}_{st}^{\text{rw}}) = \sum_{\wp \in \mathcal{P}_{st}} \mathbf{P}_{st}(\wp) \log(\mathbf{P}_{st}(\wp) / \mathbf{P}_{st}^{\text{rw}}(\wp)), \quad (1.7)$$

which is constrained to a fixed value $J_0 > 0$, and which determines the degree of randomness associated with the movement behaviour of the agent. A low value of J_0 constrains the distribution \mathbf{P}_{st} to remain very similar to the random walk distribution $\mathbf{P}_{st}^{\text{rw}}$, while a high value of J_0 allows the distribution to focus more on optimal, low-cost paths.

The solution of (1.6), which can be obtained with the Lagrangian method (see e.g. [16]), is the Gibbs–Boltzmann distribution (which we also refer to as the *RSP distribution*) over the set \mathcal{P}_{st} :

$$\mathbf{P}_{st}(\wp) = \frac{\mathbf{P}_{st}^{\text{rw}}(\wp) \exp(-\beta \tilde{c}(\wp))}{\sum_{\wp' \in \mathcal{P}_{st}} \mathbf{P}_{st}^{\text{rw}}(\wp') \exp(-\beta \tilde{c}(\wp'))}, \quad (1.8)$$

where $\beta = 1/T > 0$ is the *inverse temperature parameter*, resulting from introducing T as the Lagrangian multiplier of the relative entropy constraint. The parameter β is thus related to the relative entropy value J_0 . Namely, for low values of β (corresponding to low values of J_0), that is, when $\beta \rightarrow 0^+$, the distribution

converges to the random walk distribution. For high values of β (high values of J_0), that is, when $\beta \rightarrow \infty$, the distribution focuses more and more on the low-cost paths.

Among the other interpretations of the RSP framework, the RSP distribution could be obtained from an inverse point-of-view as well, namely by considering minimization of relative entropy with a fixed expected cost. That way the RSP framework can be interpreted as a maximum entropy model [24], as minimizing relative entropy with respect to the random walk distribution is conceptually close, though not equivalent, to maximizing the Shannon entropy.

Normally in the RSP model, for convenience, the user is required to set the value of β , instead of J_0 , to determine the degree of randomness associated to the distribution. Note that there is no analytical expression for computing the value of β that would result in the RSP distribution with a given relative entropy J_0 . Instead, for obtaining the RSP distribution with a particular relative entropy, the corresponding value of β can be obtained with a bisection search. Also worth noting is that Equation (1.8) is the solution of the problem in Equation (1.6) only if $J_0 < J_{\max}$, where J_{\max} is the relative entropy at the limit $\beta \rightarrow \infty$, where the RSP distribution is focused only on the shortest paths and zero for other paths.

Let us denote the numerator in (1.8), that is, the *likelihood of path* \wp , as

$$\tilde{w}(\wp) = \mathbf{P}_{st}^{\text{rw}}(\wp) \exp(-\beta \tilde{c}(\wp)). \quad (1.9)$$

In fact, we make this definition for all paths between s and t , not only the hitting ones. The denominator in Equation (1.8) is the *partition function* of hitting s - t -paths, which accumulates the overall likelihood of hitting s - t -paths, and which is denoted by

$$\mathcal{Z}_{st} = \sum_{\wp \in \mathcal{P}_{st}} \mathbf{P}_{st}^{\text{rw}}(\wp) \exp(-\beta \tilde{c}(\wp)) = \sum_{\wp \in \mathcal{P}_{st}} \tilde{w}(\wp). \quad (1.10)$$

Finally, by defining the *edge likelihoods* by

$$w_{ij} = p_{ij}^{\text{rw}} \exp(-\beta c_{ij}), \quad (1.11)$$

the path likelihood is, in fact, the product of the edge likelihoods along the path:

$$\tilde{w}(\wp) = \prod_{l=1}^{L(\wp)} w_{\wp^{(l-1)}, \wp^{(l)}}. \quad (1.12)$$

1.2.3 Computation of main quantities Here, we gather results from earlier literature for computing quantities from the RSP distribution that are relevant for the current work. We first present the computation of the path likelihoods \tilde{w} and the partition function, \mathcal{Z}_{st} , based on matrix computations, and then show how, using the partition function, we may compute the *expected cost* of paths from s to t , as well as the *expected number of traversals over edges* and the *expected number of visits to nodes*, over the RSP distribution. All of these quantities and their computation appear throughout the derivations related to the maximum likelihood estimation in later sections.

The edge likelihoods w_{ij} , from Equation (1.11), define the *likelihood matrix* \mathbf{W} , which can be expressed as

$$\mathbf{W} = \mathbf{P}^{\text{rw}} \circ \exp(-\beta \mathbf{C}), \quad (1.13)$$

where \mathbf{P}^{rw} and \mathbf{C} are the matrices containing the reference transition probabilities and edge costs, respectively, \circ is the element-wise, that is, Hadamard, product, and the exponential is taken element-wise as well. The likelihood matrix \mathbf{W} is substochastic, because its row sums are all less than unity, $\sum_j w_{ij} < 1$. This can be interpreted as \mathbf{W} defining a *killed random walk* (also sometimes called an *evaporating random walk*), where the residue probability at each node i , $1 - \sum_j w_{ij}$, corresponds to the probability of transition from i to an imaginary, absorbing ‘cemetery node’, instead of continuing the walk to a neighbouring node.

We present two ways of computing the partition function of Equation (1.10). The first way is to define matrix

$$\bar{\mathbf{W}} = \mathbf{W} - \mathbf{I}_t \mathbf{W} \quad (1.14)$$

as the matrix \mathbf{W} with row t set to zero. This is equivalent to considering a deletion of all edges leaving node t , which makes t an *absorbing* node (or state; analogously to absorbing Markov chains [25]).

The overall likelihood of hitting paths from s to t of given length $k = 1, 2, \dots$, is given by elements of the powers of $\bar{\mathbf{W}}$:

$$\sum_{\wp \in \mathcal{P}_{st}^{(k)}} \tilde{w}(\wp) = \left[\bar{\mathbf{W}}^k \right]_{st}. \quad (1.15)$$

As a result, the partition function, \mathcal{Z}_{st} , defined earlier in Equation (1.10), can be computed by summing over all path lengths k :

$$\mathcal{Z}_{st} = \left[\bar{\mathbf{W}} + \bar{\mathbf{W}}^2 + \bar{\mathbf{W}}^3 + \dots \right]_{st} = \left[(\mathbf{I} - \bar{\mathbf{W}})^{-1} - \mathbf{I} \right]_{st}. \quad (1.16)$$

However, computing quantities with the above approach can be costly, when considering different target nodes, t , as the matrix inverse appearing in Equation (1.16) has to be computed separately for each t . A second way of computing the partition function \mathcal{Z}_{st} is to first compute the *fundamental matrix of all paths*, given by

$$\mathbf{Z} = \mathbf{I} + \mathbf{W} + \mathbf{W}^2 + \mathbf{W}^3 + \dots = (\mathbf{I} - \mathbf{W})^{-1}, \quad (1.17)$$

whose elements z_{st} quantify the expected number of visits to node t before being killed (i.e. transitioning to the cemetery node) during a killed random walk starting from node s based on the substochastic transition matrix \mathbf{W} . Then, as was shown in [22, Appendix B] and in [8], the partition functions \mathcal{Z}_{st} , from Equation (1.10), for any s and t such that $s \neq t$, can be computed based on elements of matrix \mathbf{Z} of Equation (1.17) as

$$\mathcal{Z}_{st} = z_{st}/z_{tt}. \quad (1.18)$$

Note, that, for any s - t -pair, the italic z_{st} always refers to element (s, t) of matrix \mathbf{Z} , whereas the calligraphic \mathcal{Z}_{st} denotes the partition function, defined in Equation (1.10). Based on this result, as discussed in [22] and [8], for any s and t such that $s \neq t$, the partition function of hitting paths \mathcal{Z}_{st} can be shown to quantify the probability that a walker starting from node s and moving according to the substochastic transition matrix \mathbf{W} survives to node t before being killed.

Finally, the matrix whose element (s, t) contains the partition function \mathcal{Z}_{st} from Equation (1.10) for all $(s, t) \in V \times V$ can be expressed, based on Equation (1.18), as

$$\mathbf{Z}_h = \mathbf{Z}\mathbf{D}_Z^{-1} - \mathbf{I}, \tag{1.19}$$

where \mathbf{D}_Z is the $(n \times n)$ diagonal matrix of the diagonal elements of \mathbf{Z} . Although this expression is more convenient for computing the partition functions between multiple (or all) s - t -pairs at once, in this article we, however, rely more on the form in Equation (1.16), considering one target node t at a time.

Note that the non-negativity, substochasticity and irreducibility (as G is strongly connected) of \mathbf{W} imply that the spectral radius of \mathbf{W} is less than unity, $\rho(\mathbf{W}) < 1$. Also, although $\overleftarrow{\mathbf{W}}$ is not irreducible, as the graph is not strongly connected after the removal of the edges leaving t , we nevertheless have $\rho(\overleftarrow{\mathbf{W}}) < 1$, as setting a row to zero in a matrix cannot increase its spectral radius. This ensures, based on Perron–Frobenius theory, that the matrix series in Equations (1.16) and (1.17) converge and can be computed using the presented matrix inverses [26].

The partition function \mathcal{Z}_{st} can be manipulated in order to derive the computation of various meaningful quantities related to the RSP framework. In particular, the *expected number of traversals over an edge* (i, j) , when moving according to the RSP distribution over hitting paths from s to t , is given by (see e.g. [14])

$$\bar{n}_{ij}(s, t) = \sum_{\wp \in \mathcal{P}_{st}} P_{st}(\wp) n_{ij}(\wp) = -\frac{1}{\beta} \frac{\partial \log \mathcal{Z}_{st}}{\partial c_{ij}}, \tag{1.20}$$

where $n_{ij}(\wp)$ denotes the number of times edge (i, j) appears on path \wp . This derives from the fact that

$$\frac{\partial \tilde{c}(\wp)}{\partial c_{ij}} = n_{ij}(\wp). \tag{1.21}$$

Similarly, the *expected cost* of moving from s to t when moving according to the RSP distribution over hitting paths from s to t is (see e.g. [8])

$$\langle \tilde{c} \rangle_{st} = \sum_{\wp \in \mathcal{P}_{st}} P_{st}(\wp) c(\wp) = -\frac{\partial \log \mathcal{Z}_{st}}{\partial \beta}. \tag{1.22}$$

By altering the temperature, the expected cost over the RSP distribution interpolates between the least cost (when $\beta \rightarrow \infty$) and the expected hitting cost (when $\beta \rightarrow 0^+$) from s to t . Accordingly, the symmetrized version $\langle \tilde{c} \rangle_{st} + \langle \tilde{c} \rangle_{ts}$ interpolates between the least cost distance (multiplied by 2), and the commute cost distance.

Thanks to the above derivative expressions (1.20) and (1.22), the above quantities can be expressed in terms of elements of matrices \mathbf{W} (from Equation (1.13)) and \mathbf{Z} (from Equation (1.17)) as (again, see [8, 14])

$$\bar{n}_{ij}(s, t) = \left(\frac{z_{si}}{z_{st}} - \frac{z_{ti}}{z_{st}} \right) w_{ij} z_{jt} \tag{1.23}$$

and

$$\langle \tilde{c} \rangle_{st} = \sum_{(i,j) \in E} \bar{n}_{ij}(s,t) c_{ij} = \sum_{(i,j) \in E} \left(\frac{z_{si}}{z_{st}} - \frac{z_{ti}}{z_{tt}} \right) w_{ij} c_{ij} z_{jt}. \quad (1.24)$$

From (1.23), the *expected number of visits to node i* with respect to the RSP distribution over hitting paths from s to t , can be computed as (again, see [14] for details)

$$\bar{n}_i(s,t) = \sum_{j \in \text{Succ}(i)} \bar{n}_{ij}(s,t) = \left(\frac{z_{si}}{z_{st}} - \frac{z_{ti}}{z_{tt}} \right) z_{it}. \quad (1.25)$$

The quantity $\bar{n}_i = \sum_{s \in V} \sum_{t \in V} \bar{n}_i(s,t)$ was coined in [14] as the *simple RSP betweenness centrality* of node i , which interpolates between the *shortest path likelihood betweenness* (when $\beta \rightarrow \infty$), which is strongly related to the standard shortest path betweenness centrality [27], and the stationary distribution of the unbiased random walk (when $\beta \rightarrow 0^+$) on the graph.

The RSP distribution over paths from s to t can also be interpreted as defining a *biased random walk*, with new transition probabilities containing a drift towards t . The *biased transition probabilities* towards t can be obtained by using (1.23) and (1.25) as

$$p_{ij}^{(t)} = \frac{\bar{n}_{ij}}{\bar{n}_i} = \frac{w_{ij} z_{jt}}{z_{it}}, \quad (1.26)$$

for all $i \neq t$. As we consider hitting paths, $\bar{n}_t = 0$, and the biased transition probabilities are separately defined as zero for the target node, that is, $p_{ij}^{(t)} = 0$ for all j . As can be seen from Equation (1.26), the biased transition probabilities are independent of s , that is, $p_{ij}^{(t)}$ is the same for any edge (i,j) for all starting nodes s . The biased transition probabilities can be used for generating individual paths over a graph according to the RSP distribution.

1.3 Related work

The RSP framework was originally defined in order to develop distance and centrality measures on graphs for graph-based machine learning purposes [8, 15, 16]. Variants of the RSP framework have been developed for different use cases, including the sum-over-paths [28] and bag-of-paths [22] frameworks. Parameter tuning in applications of these frameworks (e.g. in clustering in [8] and semi-supervised classification in [29]) has usually been dealt with by using a held-out tuning data set, and by searching for the parameter value giving best performance on this data. In this article, we approach the parameter estimation problem from a more fundamental view. Note, however, that the setting is quite different from the previous applications of the RSP framework, and we leave for future work the investigation of using the results derived here for parameter tuning in other application areas of RSPs.

The RSP framework shares similarities with the *logit assignment model* and its different variants proposed in transportation science [30–33] and was originally inspired by such models [21]. Logit assignment models are sometimes criticized for a couple of reasons. First, they can become computationally untractable when considering cycles on the network because of infinitely cumulating costs [34]. This problem has been alleviated, for instance, by restricting the path set to ‘efficient’ paths [32], by constraining the path lengths [35], or by introducing a *discount factor* [34] that makes the model interpolate between unbiased and optimal behaviour similarly (but not equivalently) to the inverse temperature parameter

β in the RSP model. A second criticism about logit assignment is that it has the *independence from irrelevant alternatives (IIA)* property [31], essentially meaning that an agent leaving from s to t according to the logit probabilities will more likely select an edge that leads to more alternative s - t -paths, even if those paths are almost identical (i.e. irrelevant). This issue has been addressed, for instance, with different *nested* models [36, 37], where the set of paths (or *choice set*) is partitioned into nests containing irrelevant alternatives.

The logit assignment models originate from *random utility theory*, where edge costs (or utilities) are considered as a sum of a deterministic part and a random (‘error’ or ‘noise’) part, which leads to a logit distribution over paths similar to the Gibbs–Boltzmann distribution of the RSP framework (see e.g. [31]) when minimizing expected cost. In such a formulation, the expected cost actually corresponds to the concept of *free energy*, or *potential* in the RSP formalism [8, 22]. However, normally the RSP free energy is considered as the *relative* free energy by regularizing costs in terms of relative entropy (instead of the Shannon entropy) with respect to the unbiased random walk, (see Section 1.1). Thanks to this difference, the RSP model avoids the issue of intractability of computation that the logit assignment model faces with cycles on the network. In more detail, the regularization based on relative entropy results in the substochasticity of the likelihood matrix \mathbf{W} defined in Equation (1.13), which ensures that the RSP model can be computed for any positive value of the inverse temperature parameter β , without having to restrict the set of paths. The RSP model also alleviates the IIA issue in some cases, as the addition of an alternative route causes a change in the unbiased random walk, and thus the RSP probabilities.

The standard derivation of the RSP model considers cost-minimization subject to constrained relative entropy. The relative entropy can be interpreted information-theoretically as a fixed degree of informedness, or knowledge that the random walker has of the environment (compared to an unbiased random walker); in other words, the walker aims at minimizing travel cost constrained on its knowledge of the network. Such an information-theoretic interpretation can be expressed for some logit assignment models, as for example, the *rational inattention model* in [38]. On the other hand, the RSP model can also be derived in a similar fashion to the derivation of the logit assignment model, based on augmented edge costs, and minimization of expected augmented path costs. We leave these developments for future work, but mention them here to increase the motivation of using RSPs in modelling trajectory data, in addition to the discussion on the topic already in Section 1.1.

The literature behind logit assignment models is extensive, and also includes various studies for parameter estimation, also with maximum likelihood methods. The early works [39, 40] derive maximum likelihood methods for a parameter of the logit assignment model corresponding to the inverse temperature in our formulation. Their results are similar to the results derived in our work for complete trajectories, although the works only consider networks without cycles. Also, the *dispersion-constrained model* in [41] is based on a maximum-entropy approach similar to the RSP framework, and involves the estimation of a temperature parameter. Other works, such as [34, 35, 37, 42–44] focus, in various traffic-related settings, on the problem of estimating parameters on edges or parameters associated to features on edges, which in some cases corresponds to the problem of estimating the inverse temperature and in some case to the problem of estimating edge costs in the RSP framework (the latter of which is addressed only briefly in the current work, in Section 2.4). The above cited works in logit assignment and related models are in many respects very close and similar to the RSP model. However, one key difference is that, as already discussed above, the RSP model considered in this article, by maximizing relative entropy, generalizes a pure, unbiased random walk, whereas the logit assignment models can be seen as maximizing the Shannon entropy over paths. In addition, to the best of our knowledge, a similar definition and handling of data consisting of incomplete trajectories, as presented here, do not seem to appear in the logit assignment literature.

The problem of parameter estimation in the RSP framework has similarities with the more general problem of estimation of Markov chain transition probabilities from trajectory data (see e.g. [45–47]). Indeed, when assuming the RSP model, a given value of β defines the biased transition probabilities towards the target node, according to Equation (1.26), and thus estimating β implies estimating the transition probabilities. However, the general estimation of Markov chains, studied in the works cited above, does not involve assumptions on the distribution of paths on the graph, and also does not consider affinities, costs or other weights related to the transitions. Moreover, Markov chain estimation has been mostly studied for continuous-time Markov jump processes, for the estimation of the *generator matrix* of the process. Such techniques have been developed for example, for healthcare research, for studying disease progression and treatment effects, where the temporal aspect is of course vital [45]. Instead, the RSP framework is only analogous to a discrete time Markov chain, although extending the idea of RSPs to consider continuous time is a planned topic for future research.

Another problem related to the current work, especially the problem of model inference with incomplete trajectory data, is the inference of *aggregate Markov chains*, which has been applied to ion channel modelling [48]. There the state space is divided into aggregates and only the aggregate that the system is in, instead of the actual state, is observed. Aggregated Markov chains have also been only considered in the context of continuous-time chains. Aggregated Markov chains bare similarities with the study of *stochastic complementation* in discrete-time Markov chains [49].

Besides the temporal aspect, there are also other differences between the general estimation of Markov chains and the estimation of RSP parameters. Namely, the methods for Markov chain estimation normally require data containing observations at each state of the chain, as the transition probabilities for unobserved states cannot be estimated [50]. The estimation of the inverse temperature parameter in the RSP model, however, does not require this, and it can be estimated even for a sparse set of incomplete trajectories, as derived and verified with experiments in Section 3.

2. Maximum likelihood estimation with complete trajectories

The rest of the article is dedicated to deriving and verifying methods for computing the likelihood of the RSP model parameters when modelling trajectory data, and computing the maximum likelihood estimates (MLEs). The focus is mostly on the likelihood of the inverse temperature parameter β . For estimating β , the *graph structure*, that is, the edge affinities (or the reference transition probabilities) and edge costs, are always assumed to be known completely and exactly. The MLE of β is denoted in general as $\hat{\beta}_{\text{MLE}}$. However, in the expressions for the MLEs, β does not normally appear explicitly, but the expressions involve other RSP quantities, which can be computed with the methods detailed in Section 1.2.3. Accordingly, $\hat{\beta}_{\text{MLE}}$ can then be found by performing a search that satisfies the expressed MLE criterion, or by other, more direct optimization means.

This section deals with data sets of *complete trajectories* on a network, that is, where each node (and thus each edge) of the observed path has been recorded. We derive the results by first considering the simple case where all trajectories are observed to go from one starting node s to one destination node t , and then extend to trajectories between several s - t -pairs. The methods are then validated with artificial data. We also present briefly in this section a method for estimating the edge costs c_{ij} from complete trajectories, when assuming that only the affinities or reference transition probabilities are known, but otherwise the focus of the article is on the estimating β . Later, in Section 3, we tackle the problem of computing $\hat{\beta}_{\text{MLE}}$ when the data consist of incomplete trajectories, where only a part of the edges or nodes visited along each trajectory is observed.

2.1 Single source and target

Let Ω_{st} be a data set containing K fully observed hitting paths going from s to t .

THEOREM 1 Assuming independence between the trajectories, the maximum likelihood estimate of the inverse temperature, $\hat{\beta}_{\text{MLE}}$, given the data set Ω_{st} , as described above, is the value of β for which the expected cost of hitting s - t -paths over the RSP distribution, $\langle \tilde{c} \rangle_{st}$, from Equations (1.22) and (1.24), satisfies,

$$\langle \tilde{c} \rangle_{st} = \frac{1}{K} \sum_{\wp \in \Omega_{st}} \tilde{c}(\wp), \quad (2.1)$$

Proof. Based on the assumption of independence between the trajectories, the likelihood is simply the product of the individual path probabilities from Equation (1.8):

$$\mathcal{L}(\beta \mid \Omega_{st}) = \prod_{\wp \in \Omega_{st}} P_{st}(\wp) = \prod_{\wp \in \Omega_{st}} \frac{P_{st}^{\text{rw}}(\wp) \exp(-\beta \tilde{c}(\wp))}{\mathcal{Z}_{st}} \quad (2.2)$$

and the log-likelihood is

$$\log \mathcal{L} = \sum_{\wp \in \Omega_{st}} (\log P_{st}^{\text{rw}}(\wp) - \beta \tilde{c}(\wp)) - K \log \mathcal{Z}_{st}. \quad (2.3)$$

Taking the derivative of the log-likelihood with respect to β , and setting it to zero gives us the necessary optimality condition for the MLE:

$$\frac{\partial \log \mathcal{L}}{\partial \beta} = - \sum_{\wp \in \Omega} \tilde{c}(\wp) - K \frac{\partial \log \mathcal{Z}_{st}}{\partial \beta} = 0 \iff \langle \tilde{c} \rangle_{st} = \frac{1}{K} \sum_{\wp \in \Omega} \tilde{c}(\wp), \quad (2.4)$$

where we used Equation (1.22). □

Theorem 1 simply states that the likelihood is maximized by the value $\hat{\beta}_{\text{MLE}}$ for which the expected cost over the RSP distribution equals the average cost of the observed trajectories. A similar result is common in a large class of maximum entropy estimation procedures [24], whereas it is presented here for the first time in the literature related to RSPs. As discussed above, Equation (2.4) cannot be solved explicitly for β . Instead, the value $\hat{\beta}_{\text{MLE}}$ has to be estimated numerically by finding the root of the equation.

2.2 Multiple sources and targets

Let then Ω denote a data set of trajectories between different s - t -pairs, $\Omega_{st} \subset \Omega$ the set of trajectories in the data that go from a particular s to a particular t , and $K_{st} = |\Omega_{st}|$ the number of trajectories from s to t (with $K_{st} = 0$ if none of the trajectories go from s to t).

THEOREM 2 Given data set Ω , and assuming, again, independence between the trajectories, $\hat{\beta}_{\text{MLE}}$ is the value for which the RSP expected costs, $\langle \tilde{c} \rangle_{st}$, for all s and t satisfy

$$\sum_{s,t \in V} K_{st} \langle \tilde{c} \rangle_{st} = \sum_{s,t \in V} \sum_{\wp_{st} \in \Omega_{st}} \tilde{c}(\wp_{st}) \quad (2.5)$$

Proof. The likelihood, given Ω , is simply the product of the likelihoods for each s - t -pair and, accordingly, the log-likelihood can be written as

$$\log \mathcal{L}(\beta \mid \Omega) = \sum_{s,t \in V} \left(-K_{st} \log \mathcal{Z}_{st} + \sum_{\wp_{st} \in \Omega_{st}} (\log P_{st}^{\text{rw}}(\wp_{st}) - \beta \tilde{c}(\wp_{st})) \right). \quad (2.6)$$

Again, as in the proof of Theorem (1), setting the derivative to zero, we see that the MLE of β should satisfy

$$\sum_{s,t \in V} K_{st} \langle \tilde{c} \rangle_{st} = \sum_{s,t \in V} \sum_{\wp_{st} \in \Omega_{st}} \tilde{c}(\wp_{st}). \quad (2.7)$$

□

Thus, as in Theorem 1, as well as often in maximum entropy estimation methods [24], the most likely value of β corresponds to the one for which the empirical average agrees with the expected value given by the model. Likewise, as in Section 2, Equation (2.7) cannot be solved explicitly for β , but $\hat{\beta}_{\text{MLE}}$ has to be determined numerically.

2.3 Validation of $\hat{\beta}_{\text{MLE}}$ with complete trajectories

The above results and their accuracy were evaluated by generating trajectories on three artificial graphs. The first graph is a simple 20×20 grid with uniform edge costs. The second is a 20×20 Gaussian landscape depicted as a heatmap in Fig. 1A. In that graph, each node (corresponding to a pixel in the heatmap) is assigned a cost based on a mixture of Gaussian distributions on a plane overlaid on the grid. In more detail, the mixture consists of five low-cost and five high-cost Gaussian patches, which cause decrease and increase of the costs from a base value of 0.5. The edge costs on the graph are then determined by the cost assigned to the ending node of the edge. In both the uniform grid and the Gaussian landscape, each node is connected to its adjacent and diagonal neighbours and the cost of the diagonal connections is multiplied by $\sqrt{2}$. Affinities are fixed as reciprocals of costs, that is, $a_{ij} = 1/c_{ij}$, and reference transition probabilities (as explained earlier) as the normalized affinities, $p_{ij}^{\text{rw}} = a_{ij} / \sum_k a_{ik}$.

The third graph used for validating our results is a weighted directed graph generated with the Lanchichinetti–Fortunato–Radicchi (LFR) algorithm [51] designed for generating artificial complex networks with a community structure. We simply generated one LFR graph with 400 nodes forming 5 communities, with the mixing parameters for nodes and weights set to 0.2, the degree sequence exponent to 4 and the average degree of nodes set to 4. A visualization of the resulting LFR graph used in the experiments is depicted in Fig. 1B, where the node colours represent the community of each node. The weighted-network version of the LFR algorithm generates edge weights w_{ij} that reflect the strength of connection between nodes respecting the community structure. In order to study the effect of the possible independence between edge affinities and costs on the estimation results, we defined the affinities on the

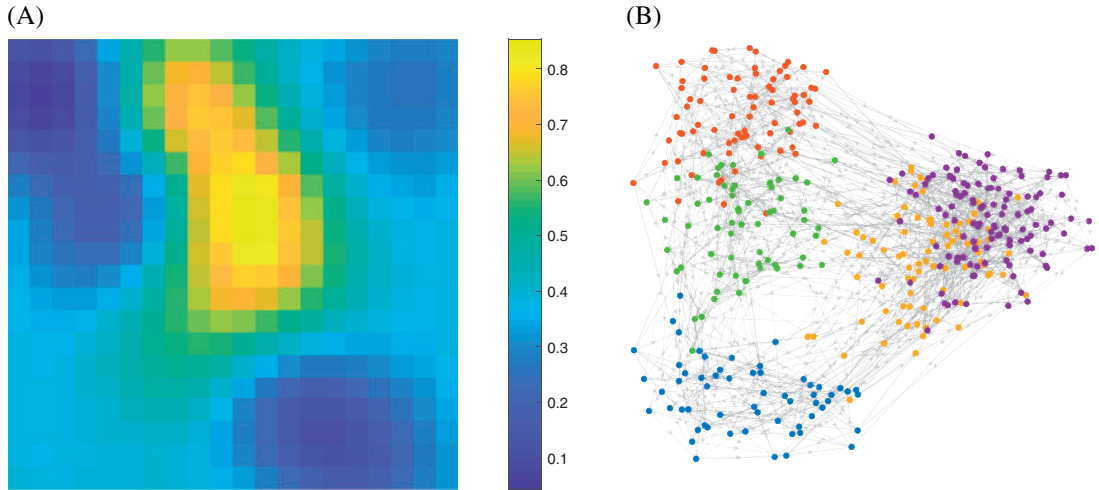


FIG. 1. Two of the three graphs used for validating the MLE methods. (A) The generated 20×20 Gaussian landscape. Edge costs are determined by costs on pixels, shown by the heatmap, so that for pixel j , $c_{ij} = c_j$ for all i s.t. $(i, j) \in E$. (B) The graph generated with the LFR algorithm. The node colours depict the nodes' communities.

LFR graph simply as $a_{ij} = 1$ for all $(i, j) \in E$, and used the weights w_{ij} generated by the LFR algorithm for defining costs by taking the reciprocal, that is, $c_{ij} = 1/w_{ij}$. Thus, considering this graph in the validation experiments, in addition to the two grid graphs introduced above, ensures that the methods developed in this work also work properly with other network structures as well as with independent edge affinities and costs.

Using a set of values for β , 200 paths were generated on both landscapes based on each studied value of β . Each path was generated by first drawing an s - t -pair uniformly randomly on the grid, however only accepting node pairs that were at least three steps apart on the grid. A path between each s - t -pair was then generated by using the biased transition probabilities from Equation (1.26). Then, for each such set of paths Ω , the value $\hat{\beta}_{\text{MLE}}(\Omega)$ was inferred by finding, by a simple line search, the value of β that satisfied Equation (2.7). The above procedure was repeated 10 times for each studied value of β and the mean and standard deviation of the MLEs was recorded. The results in Table 1 show that the MLEs, on average, indeed closely match the true values of β used for generating the paths.

2.4 Estimation of edge costs

So far, we have only focused on the problem of estimating the inverse temperature parameter β in situations where the graph structure, that is, the edge affinities (or the reference transition probabilities) and the edge costs are known a priori. In this section, however, we consider briefly the problem of estimation of edge costs from trajectories. Thus, assume that only the edge affinities (or the reference transition probabilities) are known a priori. If the affinities cannot be quantified, but the graph structure (i.e. which nodes are connected by edges) is known, then we can simply assume unit affinities. We then try to estimate the edge costs based on the observed trajectories.

It turns out that this estimation problem actually contains the problem of estimating β , meaning that solving the edge cost estimation problem also solves the estimation problem of β . This is due to the fact that concerning the RSP distribution over paths (Equation (1.8)), parameter β can be considered as a

TABLE 1 *The MLEs in the experiment with complete observed trajectories. The first column on the left shows the value of β used for generating the paths and the two other columns show the mean \pm the standard deviation of the MLEs over 10 repetitions.*

β	$\hat{\beta}_{\text{MLE}}$, uniform grid	$\hat{\beta}_{\text{MLE}}$, simulated landscape	$\hat{\beta}_{\text{MLE}}$, LFR graph
0.001	0.00096 \pm 0.00020	0.00111 \pm 0.00024	0.00113 \pm 0.00032
0.005	0.00486 \pm 0.00053	0.00526 \pm 0.00064	0.00490 \pm 0.00090
0.01	0.00970 \pm 0.00085	0.01029 \pm 0.00115	0.01056 \pm 0.00140
0.05	0.04874 \pm 0.00306	0.04894 \pm 0.00511	0.05228 \pm 0.00776
0.1	0.09785 \pm 0.00497	0.09956 \pm 0.00392	0.10351 \pm 0.01329
0.5	0.49601 \pm 0.01908	0.50897 \pm 0.02236	0.49371 \pm 0.04843
1	1.01719 \pm 0.03833	0.99922 \pm 0.02422	0.97016 \pm 0.07956
5	5.07901 \pm 0.23531	4.99453 \pm 0.17301	5.10122 \pm 0.38010
10	10.08117 \pm 1.04427	10.05533 \pm 0.35628	10.32488 \pm 0.79659

simple scaling factor of the path costs. Namely, consider a data set of trajectories is generated according to the RSP distribution with a fixed inverse temperature, say β^* , which is unknown to the user. But this is the same distribution as one would obtain by considering the RSP distribution with $\beta = 1$, but on a modified graph with edge costs $\hat{c}_{ij} = \beta^* c_{ij}$, as then, for each \wp ,

$$\beta^* \tilde{c}(\wp) = \beta^* \sum_{l=1}^{L(\wp)} c_{\wp(l-1), \wp(l)} = \sum_{l=1}^{L(\wp)} \hat{c}_{\wp(l-1), \wp(l)} = \hat{c}(\wp). \quad (2.8)$$

Expressed conversely, if we try to estimate the costs from the trajectories, we may simply assume that $\beta = 1$, as a result of which we should end up with cost estimates \hat{c}_{ij} . But these estimates contain the information of both the ‘original’ edge costs and the value β^* used to generate the data on the original graph. This shows that the estimation of β is in practice only a subproblem of the edge cost estimation problem.

We now derive the MLE of edge costs in the case of complete trajectories, given a data set Ω , as in Section 2.2. For this, we assume the value $\beta = 1$, and search for edge costs that maximize the likelihood of the trajectories.

THEOREM 3 Given data set $\Omega = \cup_{s,t \in V} \Omega_{st}$, with $K_{st} = |\Omega_{st}|$, as in Section 2.2, and assuming that $\beta = 1$, the MLE of the edge cost parameters c_{ij} , for all $(i, j) \in E$ is the value for which the expected number of visits over each edge (i, j) over hitting s - t -paths, $\bar{n}_{ij}(s, t)$, (from Equations (1.20) and (1.23)) satisfy

$$\sum_{s,t \in V} K_{st} \bar{n}_{ij}(s, t) = \sum_{s,t \in V} \sum_{\wp_{st} \in \Omega_{st}} n_{ij}(\wp_{st}). \quad (2.9)$$

Proof. Recall the log-likelihood of observing a set Ω_{st} of K trajectories from s to t , from Equation (2.3) (now with $\beta = 1$). Using Equations (1.20) and (1.21), the partial derivative of the log-likelihood w.r.t. an

edge cost c_{ij} is

$$\frac{\partial \log \mathcal{L}}{\partial c_{ij}} = K \bar{n}_{ij}(s, t) - \sum_{\wp \in \Omega_{st}} n_{ij}(\wp), \quad (2.10)$$

where, for recollection, $n_{ij}(\wp)$ is the number of traversals over edge (i, j) along path \wp and $\bar{n}_{ij}(s, t)$ is its expectation with respect to the RSP probability distribution over hitting paths from s to t .

Setting the derivative in Equation (2.10) to zero, we see that the MLE of an edge cost c_{ij} should satisfy

$$\bar{n}_{ij}(s, t) = \frac{1}{K} \sum_{\wp \in \Omega_{st}} n_{ij}(\wp). \quad (2.11)$$

The theorem follows directly by considering the data set as $\Omega = \cup_{s,t} \Omega_{st}$, with $K_{st} = |\Omega_{st}|$ where $\Omega_{st} = \{\wp \in \Omega \mid \wp \in \mathcal{P}_{st}\}$ for all $s, t \in V$. \square

Again, as Theorems 1 and 2, this result is analogous to corresponding results in maximum entropy modelling [24], but appear here for the first time in the context of RSPs. Unfortunately, as was the case with the MLE of β , the above cannot be solved analytically for c_{ij} . Moreover, the dimension of the problem is now the number of edges, so the solution cannot be found only by a simple line search. The MLE of the edge costs can, instead, be sought for example, by performing a gradient ascent towards the direction given by the edge derivatives in Equation (2.10) above. We leave the investigation of this idea for future work, and focus in the remainder of the article on the estimation of β .

3. Maximum likelihood estimation with incomplete trajectories

This section deals with the case where data consist of *incomplete trajectories* from s to t . For a path $\wp \in \mathcal{P}_{st}$, an incomplete trajectory means a subsequence of either the edge sequence or the node sequence constituting \wp . For each such trajectory, we assume that the source and target nodes, s and t are known and t is considered as absorbing; this assumption is discussed in more depth later on.

In this section, we derive methods for computing the likelihood of the inverse temperature parameter β given an observed data set Ω of incomplete trajectories, which constitute the most relevant contribution of this work. These methods provide useful tools for many applications, because often in empirical tracking studies trajectories can only be recorded partially. We derive the likelihood computation methods from the simplest case to the general one. First, we consider the cases where only one edge or one node is observed from one trajectory, and then generalize to cases where multiple edges or multiple nodes are observed over several trajectories. We also demonstrate experimentally the behaviour of the likelihood and validity of the methods.

The methods developed here for likelihood computation for incomplete trajectories are general in the sense that they make the most naive possible assumption on the method of observation, that is, the sampling of edges or nodes along the trajectories. In more detail, we assume that the observed edges have been drawn from a uniform distribution over the actual trajectory. Although simplistic, this assumption makes the method applicable to many different scenarios. The sampling assumption is discussed in more depth in Section 3.4, where we also consider, as an alternative, that the number of observations along a trajectory is binomially distributed. However, this binomial prior is not investigated in the experiments in the article, but they rely on the uniformity assumption.

3.1 Determining s and t

In the case of incomplete trajectories, determining or inferring the starting node s and the absorbing target node t is not necessarily obvious. In some cases, it can be resolved trivially by determining the last observed node on the trajectory as the target. In others, the target node might be evident in the network; for instance, considering movement in a computer game, the target node could be determined as the state where the game is completed or won. Sometimes it can also be meaningful to select an area or a subset of nodes of the network as a set of absorbing targets. Such a strategy can be more suitable in cases where the observed incomplete trajectories are sparse.

Another option, which we also use in the experiments in Section 4, is to consider, as in the previous strategy, an area of the network as the destination of movement, but decide the target t for each trajectory as the first observed node of the trajectory within this area or subset (and neglect the remaining part of the trajectory). The motivation for this is that the random walker is assumed to move according to the RSP distribution only until it reaches a particular part of the network, after which its movement behaviour might change. This assumption evidently holds for the use case in Section 4, where the data is based on seasonal migration of wild reindeer from one habitat to another.

For more sophisticated methods, one could also incorporate the *bag-of-paths (BoP) models* [22, 29] in different ways. The standard BoP framework is simply an extension of RSPs where the Gibbs–Boltzmann distribution is defined over the set of paths between all node pairs, instead of one s - t -pair at a time. This formulation leads to the *BoP probability* distribution over all s - t -pairs, meaning the probability that a given node-pair, s and t , are the source and target nodes of a path sampled from the ‘bag of paths’. These probabilities are determined by the overall costs of paths between a given s and t , compared to the costs of paths between other node-pairs. A similar model has been considered in the context of transportation research in [52]. An even more sophisticated approach would be the *margin-constrained BoP* model [53], where two distributions on the nodes of the network, $k \in V$, are given as input; one fixes the probability that k is a source node of a path, while the other fixes the probability that k is a target node of a path. The model then computes flows based on these fixed margins and the Gibbs–Boltzmann distribution over the set of all (hitting or regular) paths from the sources to the targets. In other words, the margin-constrained BoP model is applicable when the source and target distributions can be determined from data. Finally, we leave the consideration of incorporating these ideas in the maximum likelihood model for future research.

3.2 One observed edge

Consider a situation where the data set Ω consists of only one edge (i, j) that has been observed from one trajectory from s to t , where s and t are known beforehand. In this section, we present a computable expression of the likelihood function of β in such a case, discuss the intuition behind this expression, and also present an example illustrating the behaviour of the likelihood function.

3.2.1 Derivation of the likelihood function

THEOREM 4 Let φ be a hitting path from s to t drawn from the RSP distribution (Equation (1.8)), and let i be a node sampled from a uniform distribution over the node sequence of φ . The likelihood function of β in such a situation can be computed as

$$\mathcal{L}(\beta \mid (i, j); s, t) = \frac{[\mathbf{L}_{ij}]_{s, n+t}}{\mathcal{Z}_{st}}, \quad (3.1)$$

where $[\mathbf{L}_{ij}]_{s,n+t}$ is the element $(s, n+t)$ of the $(2n \times 2n)$ matrix that can be expressed using the *matrix logarithm* [54], \mathbf{log} , as

$$\mathbf{L}_{ij} = \sum_{k=1}^{\infty} \frac{\mathbf{Q}_{ij}^k}{k} = -\mathbf{log}(\mathbf{I} - \mathbf{Q}_{ij}), \quad (3.2)$$

where \mathbf{Q}_{ij} is the $(2n \times 2n)$ block matrix

$$\mathbf{Q}_{ij} = \begin{bmatrix} \overline{\mathbf{W}} & \mathbf{W}_{ij} \\ \mathbf{0} & \overline{\mathbf{W}} \end{bmatrix}. \quad (3.3)$$

Recall that $\overline{\mathbf{W}}$ is the matrix from Equation (1.14) obtained by setting row t of matrix \mathbf{W} to zero, and \mathbf{W}_{ij} is the matrix containing value w_{ij} at element (i, j) and zero elsewhere.

Proof. Let us denote by ρ the random variable corresponding to the drawing of a trajectory from \mathcal{P}_{st} along which the edge is observed. We then make the fundamental assumption that the probability distribution of ρ is the RSP distribution (see Equations (1.8) and (1.9)):

$$\mathbf{P}(\rho = \wp) = \mathbf{P}_{st}^{\text{RSP}}(\wp) = \frac{\tilde{w}(\wp)}{\mathcal{Z}_{st}}. \quad (3.4)$$

Furthermore, let ε be the random variable corresponding to the drawing of the observed edge. As discussed above, assume that ε is sampled uniformly from the edge sequence of the observed trajectory. Then the conditional probability of drawing edge (i, j) , given a particular path \wp , is given by the number of times \wp traverses edge (i, j) , n_{ij} , divided by the total number of edges traversed along the \wp , that is, its length, $L(\wp)$:

$$\mathbf{P}(\varepsilon = (i, j) | \rho = \wp) = \frac{n_{ij}(\wp)}{L(\wp)}. \quad (3.5)$$

Thus, the joint probability of observing edge (i, j) along path \wp is

$$\mathbf{P}(\varepsilon = (i, j), \rho = \wp) = \mathbf{P}(\rho = \wp) \mathbf{P}(\varepsilon = (i, j) | \rho = \wp) = \frac{\tilde{w}(\wp) n_{ij}(\wp)}{\mathcal{Z}_{st} L(\wp)}. \quad (3.6)$$

Note that the partial derivative of the path likelihood $\tilde{w}(\wp)$ (Equation (1.9)) can be expressed, using Equation (1.21), as

$$\frac{\partial \tilde{w}(\wp)}{\partial c_{ij}} = \mathbf{P}_{st}^{\text{rw}}(\wp) \frac{\partial \exp(-\beta \tilde{c}(\wp))}{\partial c_{ij}} = -\beta \tilde{w}(\wp) n_{ij}(\wp). \quad (3.7)$$

Based on this, the probability of observing edge (i, j) , which is also the likelihood function of β , is obtained from (3.6) by marginalizing out ρ :

$$\mathcal{L}(\beta | (i, j); s, t) = \mathbf{P}(\varepsilon = (i, j); \beta) \quad (3.8)$$

$$= \sum_{\wp \in \mathcal{P}_{st}} \mathbf{P}(\varepsilon = (i, j), \rho = \wp; \beta) \quad (3.9)$$

$$= \frac{1}{\mathcal{Z}_{st}} \sum_{\wp \in \mathcal{P}_{st}} \frac{\tilde{w}(\wp) n_{ij}(\wp)}{L(\wp)} = -\frac{1}{\beta \mathcal{Z}_{st}} \frac{\partial \tilde{\mathcal{Z}}_{st}}{\partial c_{ij}}, \quad (3.10)$$

where we define a new partition function

$$\tilde{\mathcal{Z}}_{st} = \sum_{\wp \in \mathcal{P}_{st}} \frac{\tilde{w}(\wp)}{L(\wp)} = \sum_{k=1}^{\infty} \sum_{\wp_k \in \mathcal{P}_{st}^{(k)}} \frac{\tilde{w}(\wp_k)}{k}, \quad (3.11)$$

with $\mathcal{P}_{st}^{(k)}$ being the set of hitting paths from s to t of length k . Recalling Equation (1.15) and that $\bar{\mathbf{W}}$ is the matrix obtained from matrix \mathbf{W} by setting row t to zero, we see that the new partition function can be expressed using the matrix logarithm:

$$\tilde{\mathcal{Z}}_{st} = \left[\bar{\mathbf{W}} + \frac{\bar{\mathbf{W}}^2}{2} + \frac{\bar{\mathbf{W}}^3}{3} + \dots \right]_{st} = \left[-\log(\mathbf{I} - \bar{\mathbf{W}}) \right]_{st}, \quad (3.12)$$

which exists, as $\rho(\bar{\mathbf{W}}) < 1$.

Next we show how the derivative of the above quantity, needed for the computation of the likelihood (3.10), can be computed. For this, we use the shorthand notation $\partial_{c_{ij}} X = \partial X / \partial c_{ij}$ for the partial derivative of a quantity X (e.g. a function or a matrix). First of all, for any $(i, j) \in E$ such that $i \neq t$,

$$\partial_{c_{ij}} \bar{\mathbf{W}} = \left(\partial_{c_{ij}} w_{ij} \right) \mathbf{I}_{ij} = \left(\partial_{c_{ij}} p_{ij}^{\text{rw}} \exp(-\beta c_{ij}) \right) \mathbf{I}_{ij} = -\beta \mathbf{W}_{ij}, \quad (3.13)$$

where $\mathbf{I}_{ij} = \mathbf{e}_i \mathbf{e}_j^\top$ is the matrix with 1 at element (i, j) and zero elsewhere, and where we have defined $\mathbf{W}_{ij} = w_{ij} \mathbf{I}_{ij}$. We can thus write the derivative of $\bar{\mathbf{W}}^k$ as

$$\begin{aligned} \frac{\partial \bar{\mathbf{W}}^k}{\partial c_{ij}} &= (\partial_{c_{ij}} \bar{\mathbf{W}}) \bar{\mathbf{W}}^{k-1} + \bar{\mathbf{W}} (\partial_{c_{ij}} \bar{\mathbf{W}}^{k-1}) \\ &= -\beta \left(\mathbf{w}_{ij} \bar{\mathbf{W}}^{k-1} + \bar{\mathbf{W}} \left(\mathbf{w}_{ij} \bar{\mathbf{W}}^{k-2} + \bar{\mathbf{W}} \left(\mathbf{w}_{ij} \bar{\mathbf{W}}^{k-3} + \dots \right) \right) \right) \\ &= -\beta \sum_{l=1}^k \bar{\mathbf{W}}^{l-1} \mathbf{w}_{ij} \bar{\mathbf{W}}^{k-l} \triangleq -\beta \mathbf{S}_{ij}^{(k)}, \end{aligned} \quad (3.14)$$

where we have defined, for convenience,

$$\mathbf{S}_{ij}^{(k)} = \sum_{l=1}^k \bar{\mathbf{W}}^{l-1} \mathbf{w}_{ij} \bar{\mathbf{W}}^{k-l}. \quad (3.15)$$

This matrix can be computed with the help of the auxiliary $(2n \times 2n)$ block matrix

$$\mathbf{Q}_{ij} = \begin{bmatrix} \mathbf{W}^{-t} & \mathbf{W}_{ij} \\ \mathbf{0} & \mathbf{W}^{-t} \end{bmatrix}, \quad (3.16)$$

as the k th power of this matrix can be shown to be:

$$\mathbf{Q}_{ij}^k = \begin{bmatrix} \mathbf{W}^{-t k} & \mathbf{S}_{ij}^{(k)} \\ \mathbf{0} & \mathbf{W}^{-t k} \end{bmatrix}. \quad (3.17)$$

Accordingly, the derivative of the new partition function, required in Equation (3.10), can be computed as

$$\frac{\partial \tilde{Z}_{st}}{\partial c_{ij}} = \mathbf{e}_s^T \sum_{k=1}^{\infty} \left(\frac{\partial \mathbf{W}^{-t k} / k}{\partial c_{ij}} \right) \mathbf{e}_t \quad (3.18)$$

$$= -\beta \mathbf{e}_s^T \sum_{k=1}^{\infty} \frac{\mathbf{S}_{ij}^{(k)}}{k} \mathbf{e}_t \quad (3.19)$$

$$= -\beta \mathbf{e}_s^T \sum_{k=1}^{\infty} \frac{\mathbf{Q}_{ij}^k}{k} \mathbf{e}_{n+t} \quad (3.20)$$

$$= -\beta \mathbf{e}_s^T \mathbf{L}_{ij} \mathbf{e}_{n+t} \quad (3.21)$$

$$= -\beta [\mathbf{L}_{ij}]_{s,n+t}, \quad (3.22)$$

where

$$\mathbf{L}_{ij} = \sum_{k=1}^{\infty} \frac{\mathbf{Q}_{ij}^k}{k} = -\log(\mathbf{I} - \mathbf{Q}_{ij}). \quad (3.23)$$

Finally, combining (3.10) and (3.22), the likelihood function can be expressed as

$$\mathcal{L}(\beta \mid (i, j); s, t) = -\frac{1}{\beta Z_{st}} (-\beta [\mathbf{L}_{ij}]_{s,n+t}) = \frac{[\mathbf{L}_{ij}]_{s,n+t}}{Z_{st}}. \quad (3.24)$$

□

It is possible to derive the derivative with respect to β of the above likelihood function. However, the form of the derivative is rather complicated, involving the matrix logarithm of a $(4n \times 4n)$ block matrix. More importantly, the root of the derivative cannot be solved in closed form, unlike in the case with complete trajectories in Section 2. Thus, for finding $\hat{\beta}_{\text{MLE}}$, it is more straightforward to search directly for a value of β that maximizes (3.1).

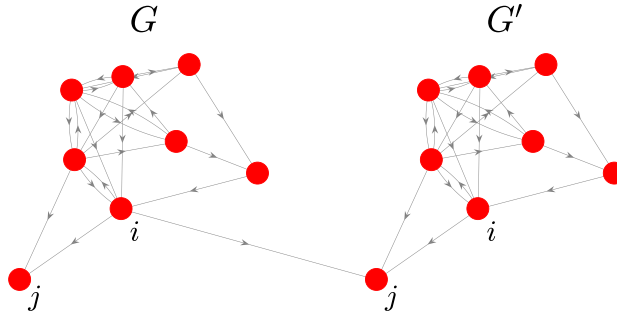


FIG. 2. The computation of the one-edge likelihood can be interpreted as making a copy G' of the original graph G and adding an edge from node i of G to node j of G' and considering likelihoods of paths from node s of G to node t of G' .

3.2.2 Intuition behind the computation of the likelihood function The computation of the single-edge likelihood can also be understood from another, more informal, perspective. Namely, the matrix \mathbf{Q}_{ij} in Equation (3.3) can be interpreted as a likelihood matrix defining a new graph consisting of the original graph G (with links going out of the target node t deleted) augmented with its copy G' and with a directed edge from node i of subgraph G to node j of subgraph G' (i.e. node $n + j$ of the new graph) with edge likelihood w_{ij} . This idea is illustrated in Fig. 2. Element $(s, n + t)$ of the k th power of matrix \mathbf{Q}_{ij} thus enumerates all s - t -paths of length k that traverse edge (i, j) and cumulates the corresponding path likelihoods.

Concerning the computation in practice, the current standard for matrix logarithm computations is the algorithm developed in [55], which is implemented, for instance, in Matlab and SciPy as the `logm` function. However, we are only interested in computing one element of the matrix logarithm, and for this purpose, we found the Matlab implementation numerically too unreliable. More exactly, the algorithm produces inaccurate values of the element $(s, n + t)$ of matrix \mathbf{L}_{ij} (see Equation (3.22)) when β is relatively large, causing the values $[\mathbf{L}_{ij}]_{s, n+t}$ to be very small. To get more reliable values, we compute the matrix logarithm element iteratively based on the power series expression of the logarithm, by computing a finite sum according to Equation (3.12).

For future research, it is worth mentioning that the above computational techniques involving the matrix logarithm could also be used, for example, for defining a network centrality measure based on the probability of uniformly sampling an edge or node over paths between nodes. This could be done considering the RSP distribution, but also with the natural random walk distribution. Namely, for the observation probability over natural random s - t -walks, one can simply perform the above computations by setting $\beta = 0$, which corresponds to replacing the matrix $\bar{\mathbf{W}}$ with the matrix $\mathbf{P}^{\text{rw}} - \mathbf{I}_t \mathbf{P}^{\text{rw}}$, that is, the random walk transition probability matrix with row t set to zero.

3.2.3 Example of the one-edge likelihood We demonstrate the behaviour of the one-edge likelihood, expressed in Equation (3.1), with a simple example illustrated in Fig. 3A. We consider a 5×5 grid with links to diagonal neighbours included. The edge costs are 1 for horizontal and vertical edges, and $\sqrt{2}$ for diagonal edges and affinities as inverse costs, $a_{ij} = 1/c_{ij}$. We consider paths from node 7, in the lower left corner, to node 25 in the upper and right-most corner. We then compute, for different values of β , the likelihood of observing each of the edges leaving from node 7 as an edge of such a path.

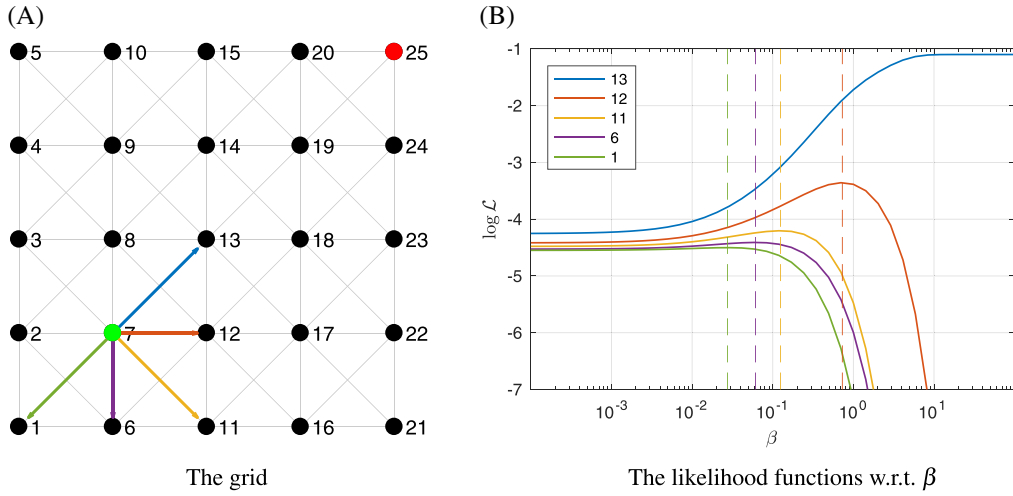


FIG. 3. The 5×5 grid, where the green node 7 is the starting node, s , the red node 25 is the absorbing target node, t (A); the log-likelihoods of observing each of the coloured edges for different values of β (B). Note the logarithmic scale on the horizontal axis.

The results are plotted in Fig. 3B, which shows the likelihoods for each edge as a function of β as separate curves. Moreover, the dashed vertical lines indicate the peaks of the curves, i.e. the value $\hat{\beta}_{\text{MLE}}$ corresponding to observing the edge in question. As is expected, the highest $\hat{\beta}_{\text{MLE}}$ value is obtained for the edge (7, 13), which lies on the least cost path from node 7 to node 25.¹ In addition, the $\hat{\beta}_{\text{MLE}}$ values decrease as we consider edges that move more and more away from the shortest path.

3.3 One observed node

The likelihood of β can also be computed in the case where a trajectory is observed to visit a particular node instead of an edge. We derive the *one-node likelihood* using the fact that the number of visits to a node i along a path \wp , $n_i(\wp)$, can be expressed as the sum of travelsals out of that node: $n_i(\wp) = \sum_{j \in \text{Succ}(i)} n_{ij}(\wp)$. Note that this definition of $n_i(\wp)$ directly means that the observed node cannot be the absorbing terminal node t , that is, that the observation is made before the end of the path.

However, the possibility that the intermediate node, drawn from a path $\wp \in \mathcal{P}_{st}$, is actually node s , must be considered more carefully. In this work, we assume that the outcome of the observation can indeed be node s , but that the observation is done *after the first step*, that is, that the observation is made from the subsequence $\wp(1, \dots, L(\wp) - 1)$, excluding the first node $\wp(0) = s$ and the last node $\wp(L(\wp)) = t$. Moreover, we assume that the node is sampled uniformly from this subsequence. The exclusion of the first node $\wp(0)$ from consideration complicates the derivation a bit compared to the derivation of the single-edge likelihood. Because of this, the full derivation of the result is presented separately in Appendix A.

¹ In fact, in this case, the likelihood can be seen to increase indefinitely, indicating that $\hat{\beta}_{\text{MLE}} = \infty$. Indeed, as β increases, the RSP probability of the least cost path along edge (7, 13) increases, and the probability of observing the edge (7, 13) increases likewise. This also leads to the increase of the likelihood function indefinitely as β increases.

THEOREM 5 Let \wp be a hitting path from s to t drawn from the RSP distribution (Equation (1.8)), and let i be a node sampled from a uniform distribution over the node sequence of \wp . Then, the likelihood function of β can be computed as

$$\mathcal{L}(\beta \mid i; s, t) = \frac{1}{Z_{st}} \sum_{u \in \text{Succ}(s)} w_{su} [\mathbf{L}_i]_{u, n+t}, \quad (3.25)$$

which contains, again, a matrix logarithm, $\mathbf{L}_i = -\mathbf{log}(\mathbf{I} - \mathbf{Q}_i)$, where

$$\mathbf{Q}_i = \begin{bmatrix} \overset{-t}{\mathbf{W}} & \mathbf{W}_{r(i)} \\ \mathbf{0} & \overset{-t}{\mathbf{W}} \end{bmatrix}, \quad (3.26)$$

and $\mathbf{W}_{r(i)} = \mathbf{I}_i \mathbf{W}$ is the matrix containing the i th row of matrix \mathbf{W} on its i th row, but zeros elsewhere.

Proof. In Appendix A. □

3.4 Multiple observed edges

In many data sets of incomplete trajectories, the trajectories contain several observations, meaning that the trajectories have been observed to pass through more than only one edge or node of the network. Here, the computation of the likelihood of β given such a sequence of multiple observed edges is presented. For this, consider a data set containing one sequence $\tilde{e} = (e_1, \dots, e_M)$ of M edges observed from one trajectory (in the corresponding order). Let us also denote $(i_m, j_m) = e_m$ for all $m = 1, \dots, M$ and assume that all i_m and j_m (except possibly j_M) are different from the target node t .

One main difference with the results presented earlier, for the one-edge and one-node likelihoods, is that now the number of observations from a trajectory also needs to be considered as a random variable. Let us denote this random variable by μ . Furthermore, let $\tilde{\varepsilon} = (\varepsilon_1, \dots, \varepsilon_M)$ be the random vector corresponding to the observed edge sequence of given length M consisting of the random variables corresponding to the observed individual edges. Similar to the uniformity assumption used in the cases of observing only one edge or node, we also assume a uniform distribution both for μ and for $\tilde{\varepsilon}$, in addition to which we assume independence between the random variables $\varepsilon_1, \dots, \varepsilon_M$. One justification for considering the uniform distribution in both cases is that they are, in this setting, the maximum entropy distributions, which implies that these assumptions are the most naive and generic ones possible.

In Section 3.7, we will also briefly consider an alternative where μ is binomially distributed. Assuming other distributions may also make the model more accurate in cases where the actual sampling is known to behave in a certain way, but the naive uniformity assumption can be useful when the sampling process is not known or is very irregular. Such is the case, for instance, with the real data example in Section 4, where we estimate β when fitting the RSP model to a data set of incomplete trajectories of wild reindeer. In that data, although the locations of the animals are mostly measured at constant time intervals, there are also missing observations causing long gaps between measurements.

Let us now derive the likelihood using the uniformity assumptions. Thus, assume that the number of observed edges, given a path \wp , is uniformly distributed, that is,

$$P(\mu = M \mid \rho = \wp) = 1/L(\wp). \quad (3.27)$$

Similarly, assume that given a path \wp and a number of observations M the probability of observing the edge sequence \tilde{e} of M edges from \wp is uniformly distributed over all $\binom{L(\wp)}{M}$ possible subsequences of M edges that can be drawn from \wp . For this, we denote by $n_{\tilde{e}}(\wp)$ the number of times that the edge sequence of \wp contains \tilde{e} as a subsequence. Then, the distribution of \tilde{e} , conditional on the number of observations, is

$$P(\tilde{e} = \tilde{e} \mid \mu = M, \rho = \wp) = \frac{n_{\tilde{e}}(\wp)}{\binom{L(\wp)}{M}}. \quad (3.28)$$

THEOREM 6 Let \wp be a hitting path from s to t drawn from the RSP distribution (Equation (1.8)), and let \tilde{e} be a sequence of M edges, sampled from a uniform distribution over the edge sequence of \wp , where, furthermore, M is drawn from a uniform distribution over $\{1, \dots, L(\wp)\}$. Then, the likelihood function of β can be computed as

$$\mathcal{L}(\beta \mid \tilde{e}; s, t) = \frac{[\mathbf{L}_{\tilde{e}}]_{s, Mn+t}}{\mathcal{Z}_{st}}, \quad (3.29)$$

where

$$\mathbf{L}_{\tilde{e}} = \sum_{k=M}^{\infty} \frac{\mathbf{Q}_{\tilde{e}}^k}{\binom{k}{M} k}, \quad (3.30)$$

and

$$\mathbf{Q}_{\tilde{e}} = \begin{bmatrix} \overset{-t}{\mathbf{W}} & \mathbf{W}_{i_1 j_1} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \overset{-t}{\mathbf{W}} & \mathbf{W}_{i_2 j_2} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \overset{-t}{\mathbf{W}} & \ddots & & \vdots \\ \vdots & \vdots & \ddots & \ddots & & \mathbf{0} \\ & & & \mathbf{0} & \overset{-t}{\mathbf{W}} & \mathbf{W}_{i_M j_M} \\ \mathbf{0} & \dots & \dots & \mathbf{0} & \overset{-t}{\mathbf{W}} & \mathbf{0} \end{bmatrix}. \quad (3.31)$$

Proof. We present an exact proof of the theorem only in the special case of observing two edges in Appendix B. Below, we sketch the intuition behind the proof in an informal way. \square

Using Equations (3.27) and (3.28), the likelihood can be written as

$$\mathcal{L}(\beta \mid \tilde{e}; s, t) = P(\tilde{e} = \tilde{e}, \mu = M) \quad (3.32)$$

$$= \sum_{\wp \in \mathcal{P}_{st}} P_{st}(\wp) P(\tilde{e} = \tilde{e} \mid \mu = M, \rho = \wp) P(\mu = M \mid \rho = \wp) \quad (3.33)$$

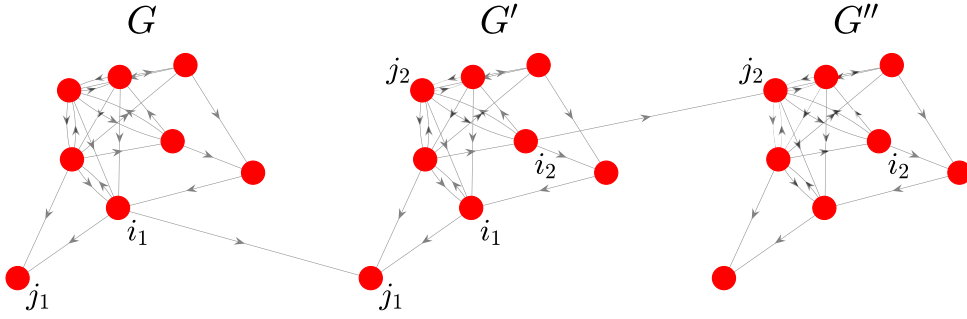


FIG. 4. The computation of the two-edge likelihood can be interpreted as making two copies, G' and G'' , of the original graph G and adding the required edges between the copies.

$$= \sum_{\wp \in \mathcal{P}_{st}} P_{st}(\wp) \frac{n_{\tilde{e}}(\wp)}{\binom{L(\wp)}{M}} \cdot \frac{1}{L(\wp)} \quad (3.34)$$

$$= \frac{1}{Z_{st}} \sum_{k=M}^{\infty} \frac{1}{\binom{k}{M} k} \sum_{\wp_k \in \mathcal{P}_{st}^{(k)}} \tilde{w}(\wp_k) n_{\tilde{e}}(\wp_k). \quad (3.35)$$

The exact manipulation of the above expression is fairly involved and is presented in Appendix B only for the case of observing two edges on a trajectory. The derivation for two edges extends naturally to the case of an arbitrary number of observed edges, although the proof of this is omitted for brevity. Here, we only give an intuitive description of the derivation of the multiple-edge likelihood based on the idea of copying the graph, used also earlier for explaining the computation of the one-edge likelihood, in Section 3.2 and illustrated in Fig. 2.

Namely, for the edge sequence \tilde{e} , we may consider producing M duplicates of the original graph G and connect the duplicates via the edges of the sequence \tilde{e} . This is illustrated for the case of two observed edges in Fig. 4. Indeed, the extended likelihood matrix $\mathbf{Q}_{\tilde{e}}$ from Equation (3.31) represents such an extended graph of $M + 1$ copies of the original graph.

Now, it can be shown that the inner sum in the expression in (3.35) is given by element $(s, Mn + t)$ of the k th power of the matrix $\mathbf{Q}_{\tilde{e}}$; more exactly

$$\sum_{\wp_k \in \mathcal{P}_{st}^{(k)}} \tilde{w}(\wp_k) n_{ij}(\wp_k) = [\mathbf{Q}_{\tilde{e}}^k]_{s, Mn+t}. \quad (3.36)$$

This result is proved formally for the two-edge likelihood in Appendix B. The case of observing M edges follows from there naturally. Inserting the above Equation (3.36), as well the definition in Equation (3.30), into Equation (3.35) leads to the desired result.

Note that unlike in the one-edge or one-node likelihood cases, the expression of the likelihood in Theorem 6 has no direct connection with the matrix logarithm, and that the expression in Equation (3.30) cannot be written in closed form. Instead, we compute the likelihood by computing the sum in Equation (3.30) up to convergence.

3.5 Multiple observed nodes

The likelihood of β given an incomplete node-trajectory can be computed similarly to the likelihood of the incomplete edge-trajectory. The derivation of the computation is similar to the extension from the one-edge likelihood to the one-node likelihood presented in Appendix 3.3. Because of this, we omit the derivation and only state the result.

THEOREM 7 Again, let \wp be a hitting path from s to t drawn from the RSP distribution (Equation (1.8)), and now, let \tilde{v} be a sequence of M nodes, sampled from a uniform distribution over the node sequence of \wp (excluding the first and last nodes of the sequence),² where, furthermore, M is drawn from a uniform distribution over $\{1, \dots, L(\wp) - 1\}$. With these assumptions, the multiple-node likelihood is given, similarly to Equations (3.25) and (3.29), by

$$\mathcal{L}(\beta \mid \tilde{v}) = \frac{1}{Z_{st}} \sum_{h \in \text{Succ}(s)} w_{su} [\mathbf{L}_{\tilde{v}}]_{h, Mn+t}, \quad (3.37)$$

where the matrix $\mathbf{L}_{\tilde{v}}$ is defined equivalently to Equation (3.30), but by replacing matrix $\mathbf{Q}_{\tilde{e}}$ with matrix $\mathbf{Q}_{\tilde{v}}$, which, furthermore, is defined equivalently to the definition of the matrix $\mathbf{Q}_{\tilde{e}}$ in Equation (3.31), however, replacing the matrices $\mathbf{W}_{i_m j_m}$ with the node-based matrices $\mathbf{W}_{r(i_m)} = \mathbf{I}_{i_m i_m} \mathbf{W}$, which contain row i_m of \mathbf{W} on row i_m and zeros elsewhere (see Equation (3.26)).

Proof. This can be shown by extending the proof of Theorem 6 similarly to the extension from the one-edge likelihood in Theorem 4 to the one-node likelihood in Theorem 5. However, the presentation is omitted from here for brevity. \square

The next section is devoted to verifying the above result with an artificial experiment, and in Section 4 its functionality and usability is tested with real data of incomplete trajectories of wild reindeer.

3.6 Validation of $\hat{\beta}_{\text{MLE}}$ with incomplete trajectories

We evaluated the applicability and precision of the MLE method in the case of incomplete trajectories from s to t . We only considered incomplete node trajectories and the corresponding method for computing the MLE. The setting was similar to the experiment for validating the MLEs for complete trajectories in Section 2. Namely, we again generated 200 paths for different values of β between uniformly distributed s - t -pairs, where s and t were at least 3 steps apart from each other. We used the same three graphs that were used in Section 2, that is, a grid with uniform costs, a simulated Gaussian landscape and a graph generated with the LFR algorithm.

From the 200 generated paths, we extract a set Ω of 200 incomplete node trajectories by sampling M nodes, where $M = \min(300, M')$, and M' is drawn uniformly from $1, \dots, L(\wp)$. We limit the maximum length of an incomplete trajectory to 300 for computational efficiency. We then compute $\hat{\beta}_{\text{MLE}}$ by

² Again, as in the one-node likelihood case, studied in Section 3.3, the possibility of the first node being s must be considered separately. As was done there, we here also make the assumption that the first observed node can be s , but it cannot be the first node, $\wp(0)$, of the observed path \wp .

TABLE 2 The MLEs in the experiment with incomplete node trajectories. The first column on the left shows the value of β used for generating the paths and the two other columns show the mean \pm the standard deviation of the MLEs over 10 repetitions.

β	$\hat{\beta}_{\text{MLE}}$, uniform grid	$\hat{\beta}_{\text{MLE}}$, simulated landscape	$\hat{\beta}_{\text{MLE}}$, LFR graph
0.001	0.00101 \pm 0.00016	0.00106 \pm 0.00013	0.00108 \pm 0.00033
0.005	0.00497 \pm 0.00043	0.00510 \pm 0.00069	0.00471 \pm 0.00119
0.01	0.00980 \pm 0.00070	0.00992 \pm 0.00088	0.01045 \pm 0.00198
0.05	0.05014 \pm 0.00275	0.05091 \pm 0.00250	0.05297 \pm 0.00860
0.1	0.10117 \pm 0.00704	0.09433 \pm 0.00807	0.10384 \pm 0.01226
0.5	0.50810 \pm 0.03167	0.49411 \pm 0.02237	0.49841 \pm 0.05781
1	1.01074 \pm 0.07147	0.98349 \pm 0.03991	0.98186 \pm 0.07691
5	4.92557 \pm 0.27878	4.93601 \pm 0.24915	5.09714 \pm 0.45950
10	12.73153 \pm 5.07237	10.08324 \pm 0.28333	10.49840 \pm 0.98032

performing a line search on β for finding the maximum value of $\log \mathcal{L}(\beta | \Omega)$. This is again repeated 10 times and we report the mean and standard deviation of the 10 obtained MLE values.

The results are gathered in Table 2. They show that the MLEs are accurate throughout the tested range of values of β and thus confirm that the methodology derived in Section 3 can be used in practice. Moreover, quite surprisingly, the results are almost as accurate as the results in Table 1 in the experiment with complete trajectories of Section 2.3.

3.7 Binomial distribution on number of observations

So far, we assumed that the number of observations in an incomplete trajectory is a random variable distributed uniformly over the range $[1, L(\wp)]$, where \wp is the trajectory being observed. In this section, we briefly discuss an alternative for the uniformity assumption by considering a binomial distribution over the number of observations, $\mu \sim \text{Bin}(L(\wp), p_\mu)$ with the condition that $L(\wp) > 0$. We only derive here a way for computing the likelihood from incomplete trajectories with this assumption, but leave further examination of this idea for future work.

Considering that the random variable corresponding to the number of observations of an incomplete trajectory, μ , is binomially distributed, involves an additional parameter, p_μ , that is, the probability of making an observation at each step of an observed path. Thus, using this assumption means that p_μ can be estimated from the data by some means. Assuming a binomial distribution for μ , the probability of observing $M \geq 1$ edges from a path \wp is

$$\begin{aligned}
 \text{P}(\mu = M | \mu \geq 1) &= \frac{\text{P}(\mu = M)}{1 - \text{P}(\mu = 0)} \\
 &= \frac{1}{1 - q_\mu^M} \binom{L(\wp)}{M} p_\mu^M q_\mu^{L(\wp)-M}, \\
 &= \frac{p_\mu^M q_\mu^{L(\wp)}}{q_\mu^M (1 - q_\mu^M)} \binom{L(\wp)}{M},
 \end{aligned} \tag{3.38}$$

where p_μ is a fixed probability that an observation is recorded at any given step of path of \wp and $q_\mu = 1 - p_\mu$. Inserting this to the expression of the likelihood in Equation (3.33) gives

$$\mathcal{L}(\beta \mid \tilde{e}) = \frac{p_\mu^M}{q_\mu^M(1 - q_\mu^M)} \sum_{\wp \in \mathcal{P}_{st}} P_{st}(\wp) n_{\tilde{e}}(\wp) q_\mu^{L(\wp)}, \quad (3.39)$$

which can be computed by replacing the definition of $\mathbf{L}_{\tilde{e}}$, instead of Equation (3.30), as

$$\mathbf{L}_{\tilde{e}} = \sum_{k=1}^{\infty} (q_\mu \mathbf{Q}_{\tilde{e}})^k = (\mathbf{I} - q_\mu \mathbf{Q}_{\tilde{e}})^{-1}. \quad (3.40)$$

We only need the element $[\mathbf{L}_{\tilde{e}}]_{s, Mn+t}$ from this matrix, which can be obtained, for instance, by solving the $(Mn + t)$ th column, $\mathbf{I}_{Mn+t}^{\tilde{e}}$, of $\mathbf{L}_{\tilde{e}}$ from the linear system

$$(\mathbf{I} - q_\mu \mathbf{Q}_{\tilde{e}}) \mathbf{I}_{Mn+t}^{\tilde{e}} = \mathbf{e}_{Mn+t} \quad (3.41)$$

and picking its s th element. Then, the likelihood is given by replacing Equation (3.29) with

$$\mathcal{L}(\beta \mid \tilde{e}) = \frac{p_\mu^M}{q_\mu^M(1 - q_\mu^M)} \cdot \frac{[\mathbf{L}_{\tilde{e}}]_{s, Mn+t}}{\mathcal{Z}_{st}}. \quad (3.42)$$

Computing $[\mathbf{L}_{\tilde{e}}]_{s, Mn+t}$ from the matrix inverse in Equation (3.41) can be done more efficiently than the computation from Equation (3.30), which cannot be expressed in closed form. However, as mentioned earlier, this method requires setting the observation probability p_μ as an additional parameter, which is not needed in the method based on the uniform distribution. However, as already mentioned, in the remainder of the article, we will only focus on the uniformity assumption, as described in Section 3.4, and leave the further study of the binomial assumption, discussed here, for future work.

4. Application to animal movement modelling

In order to test the applicability of the method for computing the MLE of β in a real data setting, we ran the MLE method for a set of GPS trajectory data collected from individual wild mountain reindeer in the Austhei area of Norway. The reason for studying this data is based on the apparent suitability of RSPs for modelling animal movement. This holds especially for the study of wild reindeer, which are a migratory species, that often have a good sense of knowledge of their environment. Accordingly, they can be considered to follow fairly optimal routes when moving in a landscape. It would be, however, unrealistic to assume that the animals always follow only the optimal path on a static landscape. Instead their movement decisions may be considered to involve some randomness. This is exactly the kind of scenario that the RSP framework is designed for. However, the aim of the experiments reported here was simply to test whether the MLEs are practical to compute for real data with the methods derived above and to see whether the estimates obtained seem sensible. We leave a more careful analysis of the obtained results, and their applicability in more focused ecological problems, such as actual prediction of movement, for future work.

The data considered here consist of incomplete trajectories recorded during springtime migration, when the reindeer traverse the landscape from north to south crossing over a road passage cutting through

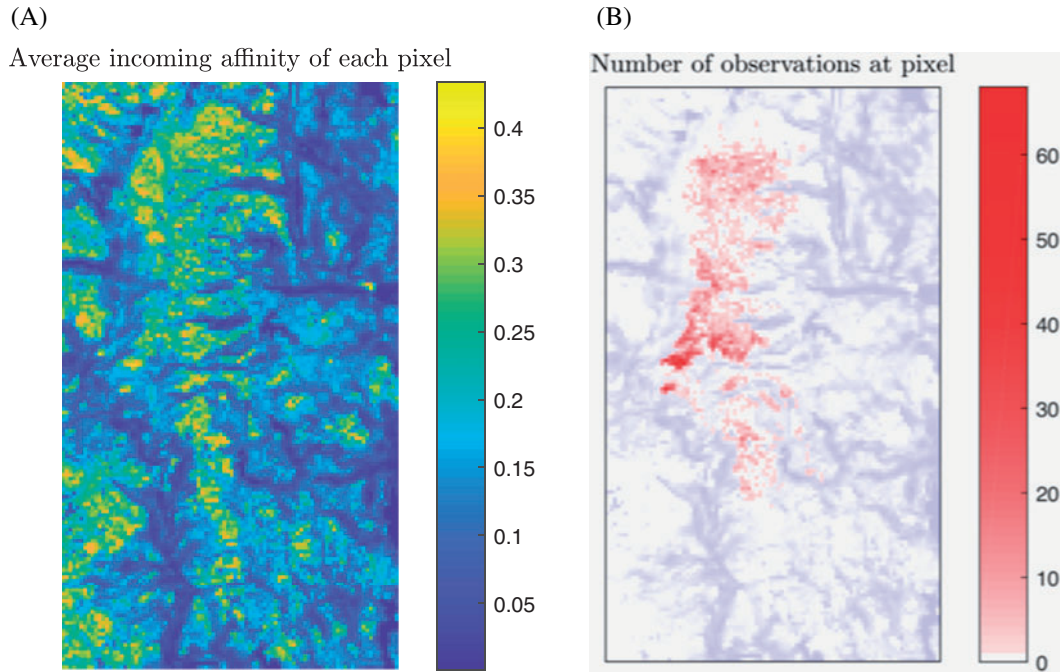


FIG. 5. Visualization of the landscape used for studying wild reindeer movement showing the average incoming edge affinity of each pixel as a heatmap (A). The heatmap of the number of times an animal was observed at each pixel (in red hue; the blue hue shows the low-affinity (i.e. high cost) pixels simply in order to display the shape of the landscape on the image) based on the GPS data (B).

the landscape. The same area and partly the same data were studied previously in the context of RSPs in [19], although there the graph was constructed differently compared to the experiment presented here. In [19], a method was also devised for estimating the inverse temperature parameter, however based on a very different, more situation-dependent approach, compared to the more generic method introduced in this work.

The landscape is modelled as a rectangular grid graph, where each node represents a 500 m wide square pixel of a map image of the landscape. The size of the landscape is 80.5 km \times 46 km, resulting in a graph with 14812 nodes. As before, each pixel was again connected to its eight surrounding pixels. First, the edge affinities a_{ij} were defined as Step Selection Probabilities (SSP) [56], inferred by detecting features on each pixel with remote sensing methods and by measuring the preference of those features for movement based on the reindeer GPS data. A more detailed explanation of how the affinities were defined is provided in Appendix C. The edge costs were defined simply as the reciprocals of affinities, i.e. $c_{ij} = 1/a_{ij}$. The landscape is illustrated in Fig. 5A as a heatmap, where the pixels are coloured according to the average incoming edge affinity of each pixel.

The data contain 32 incomplete trajectories recorded during the years 2007–2013 from 14 different individuals. Figure 5B visualizes the trajectory data by marking the number of times any of the 32 trajectories were observed at each pixel. The GPS measurements were made for the most part every 3 h, 50 % of trajectories contained all locations. However, many trajectories contain gaps of 6 h to 1 day, and one trajectory is missing locations for nearly 2 weeks. Trajectories started in the winter range

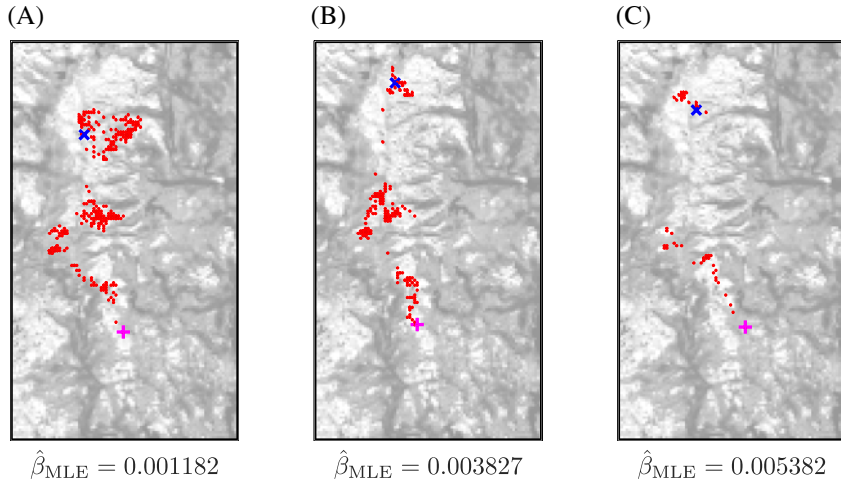


FIG. 6. Examples of three trajectories of individual reindeer and the MLE of β of each trajectory. The starting point, on the north side of the landscape, is depicted with a blue 'x', and the ending point with a magenta '+'. All 32 trajectories in the data have different source and target nodes s and t .

between 1st and 20th March, and ended in the summer range between 8th and 30th June. We cut the original trajectories after they have crossed a certain line which can be interpreted as a border of the summer range. The number of observations in the resulting trajectories was between 57 and 535. For each observation, we detect the pixel that the animal is in, and use the method derived in Section 3 for computing the multiple-node likelihood. We consider the trajectories as extracted from hitting paths, where the hitting node is selected to be the first pixel within the summer range where the animal has been observed. As discussed in Section 3.1, more sophisticated methods could be used for determining the hitting target node, such as the *margin-constrained RSP model* [53], where the graph nodes are assigned probabilities of being a starting or ending node of a path.

We first computed $\hat{\beta}_{\text{MLE}}$ for each trajectory separately, resulting in 32 MLE values. Figure 6 shows three examples of trajectories and the corresponding $\hat{\beta}_{\text{MLE}}$ values. As can be seen, the MLE values from Fig. 6A–C increase. Similarly, the directness of the trajectories, in that order, seemingly increases. The trajectory in Fig. 6A appears to circulate fairly randomly before arriving at the destination, whereas the trajectory in Fig. 6C is heads quite straightforwardly towards the destination. The trajectory in Fig. 6B appears as an intermediate type between the two others. These examples provide a sanity check and show that the MLE method gives reasonable and meaningful estimates of β .

A histogram of the MLE values of the 32 trajectories is presented in Fig. 7. The MLEs remained for the most part in a fairly consistent range of values, between $[0.001, 0.01]$, with one exceptional trajectory obtaining a MLE below this range, namely $\hat{\beta}_{\text{MLE}} = 0.0004715$. The mean of the individual MLEs was $\langle \hat{\beta}_{\text{MLE}}(\mathcal{P}) \rangle_{\mathcal{P} \in \Omega} = 0.003311$, and is represented in Fig. 7 as the dashed magenta line.

In addition to estimating the individual trajectory MLEs, we also considered the whole set of trajectories as being generated by the same value of β , and thus computed the population-wide MLE value using the whole set of trajectories. Note that all the trajectories are between different s - t -pairs, where s is simply the first observed node of the trajectory, on the winter range, and t is, as explained earlier, the first observed node on the trajectory that is located on the summer range. We consider the likelihood of the set as the product of the likelihoods of each trajectory, that is, that the trajectories are independent.

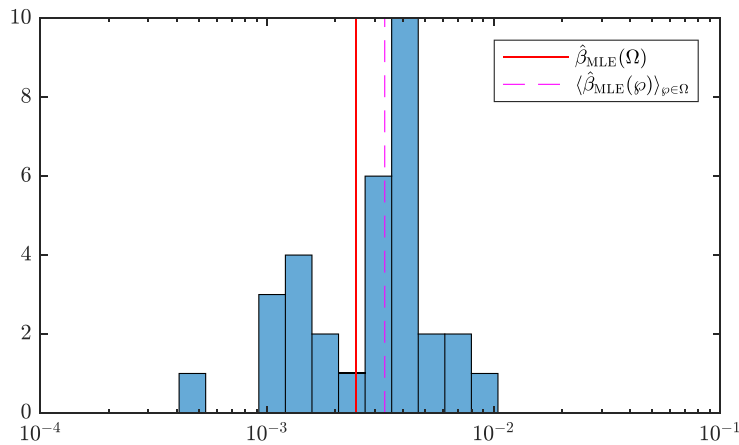


FIG. 7. Histogram of the MLE values for the 32 trajectories of reindeer. The solid red line marks the MLE obtained with the whole set of trajectories, $\hat{\beta}_{MLE}(\Omega) = 0.002466$, whereas the magenta dashed line shows the mean of the individual trajectory MLEs, $\langle \hat{\beta}_{MLE}(\varphi) \rangle_{\varphi \in \Omega} = 0.003311$.

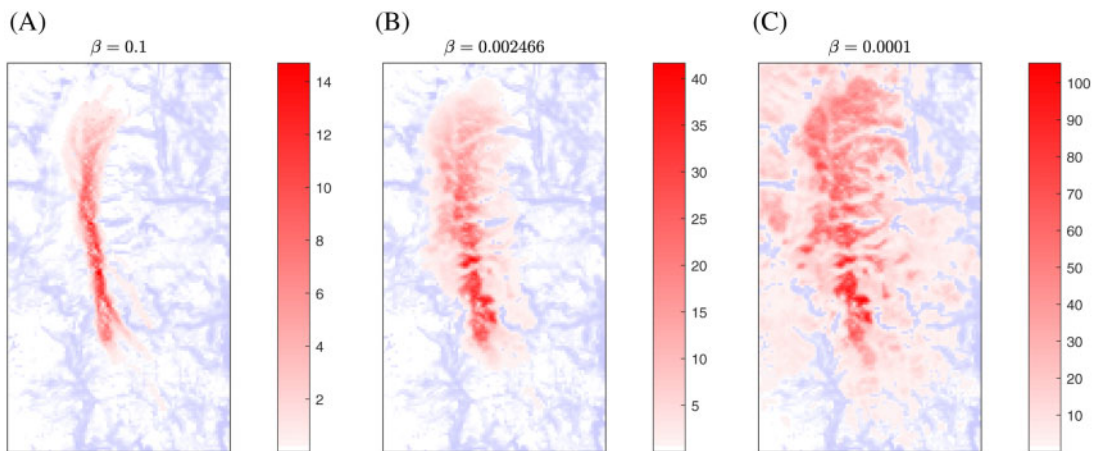


FIG. 8. The expected number of visits to each pixel over the RSP probability distributions between each observed s - t -pair, with three values of $\beta = 0.1$ (A), $\hat{\beta}_{MLE}(\Omega) = 0.002466$ (B; the MLE based on the set of trajectories Ω), and $\beta = 0.0001$ (C). The red hue at pixel i marks the value $\sum_{(s,t) \in X} \bar{n}_i(s, t)$, given the value of β depicted above the plot. Here, X denotes the set of s - t -pairs of the trajectories in the data set Ω .

Assuming independence between the trajectories is convenient for computation, but is also justified by the fact that the trajectories are mostly collected over different years and can be from individuals belonging to different herds. The MLE given by the whole set of trajectories was $\hat{\beta}_{MLE}(\Omega) = 0.002466$, and is marked by the solid red line in Fig. 7.

Figure 8 contains plots for the expected numbers of visits according to the RSP model with different values of β . These were computed by considering the starting and ending nodes, s and t , of each trajectory in the data separately, computing the expected number of visits to each pixel for that s - t -pair, and by summing the contributions of each s - t -pair. In each plot, the blue hue represents the average incoming

edge cost of each pixel, and the transparent red hue represents the observed or expected number of visits by an animal to a pixel.

Figure 8A corresponds to the model computed with $\beta = 0.1$, and Fig. 8C with $\beta = 0.0001$ for each s - t -pair. In between, Fig. 8B shows the result with the value $\beta = 0.002466$, which is the MLE estimated by using the whole set of trajectories. We also constructed another plot by using for each s - t -pair the MLE of β of the corresponding trajectory, but it is not presented here, because it resembles almost exactly the plot obtained with the global MLE in Fig. 8B.

A visual inspection of the different plots obtained with the RSP distribution using different values of β indicates that the model using the MLE values of β has most resemblance with the plot of the actual observations. Note, however, that the plots in Fig. 8 are not exactly expected to resemble the plot in Fig. 5B. Namely, as explained before, the RSP model does not take into account the temporal aspect of the movement. The temporal aspect, however, is grained in the trajectories, as the speed of movement and the time interval between observations vary. For instance, after crossing the road over to the south side, the animals tend to move faster away from the road [57]. This results generally in a fewer number of observations in the pixels south of the road. In other words, Fig. 5B rather depicts the *time spent* at each pixel by the animals. Instead, the plots in Fig. 8, given by the RSP model, reflect the importance of each pixel as an intermediate point of movement, which can be more crucial, for instance, for detecting corridors and barriers in a landscape.

5. Conclusion

This paper focused on the estimation of parameters when fitting the RSP model to data containing trajectories on a network, with most focus on the estimation of the inverse temperature parameter β . Methods were derived for computing and maximizing the likelihood of values of β given a data set of either complete or incomplete trajectories (in Sections 2 and 3, respectively). The methods were shown to provide accurate estimates of β with simulation examples where trajectories were generated on artificially constructed graphs. In addition to estimation of β , also maximum likelihood estimates of the edge costs, c_{ij} , $(i, j) \in E$, of a graph were derived, when dealing with complete trajectories.

The MLE method derived for incomplete trajectories was also tested on real data of trajectories of wild mountain reindeer on a landscape area in Norway. The purpose of the experiment was only to verify further that the MLEs of β can be computed in practice and that the estimates seem reasonable and sensible. This indicates that the MLE method provides a well-founded and functional way for model fitting, when using the RSP model for more specific applications related to animal movement.

The most significant theoretical results of the paper, in Section 3, dealt with fitting the parameter β to data consisting of incomplete trajectories. As a computational peculiarity, the derivation of the likelihood function in the special case of observing only one edge or node of a trajectory was shown to involve the matrix logarithm. In fact, the computational techniques appearing in the derivation could be applied for various other network analysis purposes. For instance, a similar method can be derived for computing the expected average edge cost of paths, which can be of use in situations where, for instance, two nodes may be considered more reliably connected when there are, on average, no high-cost edges between them. Also, as discussed in Section 3, the matrix logarithm could be used for computing the probability of observing an edge or node over the natural random walk distribution between an s - t -pair, assuming the uniform distribution over the path for the sampling process. This could be developed further to define a new network centrality measure based on the observation probability of edges or nodes.

One further extension of the theory developed in this work is to consider features on the edges or nodes of the network as separate costs, and to fit parameters on those features assuming the RSP model.

Likelihood maximization could also be developed for other observation scenarios than the one dealt in this work. One example would be a case where some of the nodes or edges of the network contain sensors and the trajectories can only be observed when they visit these sensor nodes or edges. Also, the RSP framework will in future work be extended to temporal networks, which will bring more challenges also to the parameter estimation problem. Lastly, the methods developed here will hopefully indicate new principled ways of selecting an appropriate value for β for other network data analysis tasks that the RSP framework is used for.

Acknowledgements

This work was supported by Tekniikan edistämissäätiö (The Finnish Foundation for Technology Promotion) and partially supported by the Renewable Reindeer and OneImpact projects of the Norwegian Institute for Nature Research, funded by the Research Council of Norway, as well as the Elis-IT project funded by the Région wallonne and the Brufence project funded by Innoviris (Région bruxelloise), Belgium. We thank these institutions for giving us the opportunity to conduct both fundamental and applied research. We also thank the anonymous reviewers for their relevant remarks and suggestions that helped us to improve significantly the manuscript.

Appendix: Additional material and proofs of main results

A. Proof of Theorem 5

As before, let ρ be the random variable corresponding to the drawing of a path from the set \mathcal{P}_{st} according to the RSP distribution (Equation (1.8)). Given a path $\wp = (\wp(0) = s, \wp(1), \dots, \wp(L(\wp)) = t)$, let us consider two random variables:

- ν_0 , corresponding to the drawing of an intermediate node, drawn uniformly from the subsequence of \wp excluding the last node, but *including the first node*, that is, $(\wp(0), \dots, \wp(L(\wp) - 1))$, and
- ν_1 corresponding to the drawing of an intermediate node, drawn uniformly from the subsequence of \wp *excluding both the first and last node*, that is, $(\wp(1), \dots, \wp(L(\wp) - 1))$.

Recalling the assumptions stated in Section 3.3, the one-node likelihood is given by the probability distribution of ν_1 . However, we first derive the distribution of ν_0 as an intermediate result and use it to derive the distribution of ν_1 .

Using $n_i(\wp) = \sum_{j \in \text{Succ}(i)} n_{ij}(\wp)$, as earlier, the conditional probability of node i as the outcome of ν_0 , given a path $\wp \in \mathcal{P}_{st}$, is

$$\mathbb{P}(\nu_0 = i \mid \rho = \wp) = \frac{\bar{n}_i(\wp)}{L(\wp)} = \frac{\sum_{j \in \text{Succ}(i)} \bar{n}_{ij}(\wp)}{L(\wp)}. \quad (\text{A.1})$$

Then, marginalizing out ρ , we can write the distribution of ν_0 , in similar fashion to Equation (3.10) for the one-edge likelihood, as

$$\mathbb{P}(\nu_0 = i; \beta) = \frac{1}{\mathcal{Z}_{st}} \sum_{j \in \text{Succ}(i)} \sum_{\wp \in \mathcal{P}_{st}} \frac{\tilde{w}(\wp) n_{ij}(\wp)}{L(\wp)} \quad (\text{A.2})$$

$$= -\frac{1}{\beta \mathcal{Z}_{st}} \sum_{j \in \text{Succ}(i)} \frac{\partial \tilde{\mathcal{Z}}_{st}}{\partial c_{ij}}. \quad (\text{A.3})$$

This equation is the equivalent of Equation (3.10) for edges. Now, similar to Equation (3.14), we have, for any $k \geq 1$

$$\sum_{j \in \text{Succ}(i)} \frac{\partial \bar{\mathbf{W}}^k}{\partial c_{ij}} = -\beta \sum_{j \in \text{Succ}(i)} \sum_{l=1}^k \bar{\mathbf{W}}^{l-1} \mathbf{w}_{ij} \mathbf{W}^{k-l} \quad (\text{A.4})$$

$$= -\beta \sum_{l=1}^k \bar{\mathbf{W}}^{l-1} \left(\sum_{j \in \text{Succ}(i)} \mathbf{w}_{ij} \right) \mathbf{W}^{k-l} \quad (\text{A.5})$$

$$= -\beta \sum_{l=1}^k \bar{\mathbf{W}}^{l-1} \mathbf{W}_{r(i)} \bar{\mathbf{W}}^{k-l} \quad (\text{A.6})$$

$$\triangleq -\beta \mathbf{S}_i^{(k)}, \quad (\text{A.7})$$

where $\mathbf{W}_{r(i)} = \mathbf{I}_{ii} \mathbf{W}$ is the matrix containing the i th row of matrix \mathbf{W} on its i th row, but zeros elsewhere, and $\mathbf{S}_i^{(k)}$ defined analogously to $\mathbf{S}_{ij}^{(k)}$ in Equation (3.15).

The rest of the derivation of the distribution of ν_0 proceeds as in the single-edge likelihood case, but with matrix $\mathbf{W}_{r(i)}$ in place of matrix \mathbf{W}_{ij} . Namely, the derivative expression appearing in Equation (A.3) can be computed (as in Equations (3.18)–(3.22)), as

$$\sum_{j \in \text{Succ}(i)} \frac{\partial \tilde{\mathcal{Z}}_{st}}{\partial c_{ij}} = \mathbf{e}_s^\top \sum_{k=1}^{\infty} \frac{\mathbf{S}_i^{(k)}}{k} \mathbf{e}_t, \quad (\text{A.8})$$

which, again, can be computed as element $(s, n+t)$ of the matrix logarithm $\mathbf{L}_i = -\log(\mathbf{I} - \mathbf{Q}_i)$, with the $(2n \times 2n)$ block matrix (see Equation (3.20))

$$\mathbf{Q}_i = \begin{bmatrix} \bar{\mathbf{W}} & \mathbf{W}_{r(i)} \\ \mathbf{0} & \bar{\mathbf{W}} \end{bmatrix}. \quad (\text{A.9})$$

In conclusion, we can write the distribution of ν_0 as

$$\mathbb{P}(\nu_0 = i) = -\frac{1}{\beta \mathcal{Z}_{st}} (-\beta [\mathbf{L}_i]_{s, n+t}) = \frac{[\mathbf{L}_i]_{s, n+t}}{\mathcal{Z}_{st}}. \quad (\text{A.10})$$

Using the above, we can then derive the one-node likelihood according to the distribution of ν_1 . We do this by considering a split of each path $\wp \in \mathcal{P}_{st}$ into a path consisting of the first step, from s to one of its successor nodes $u \in \text{Succ}(s)$, and the rest of the path, from u to t . We denote, generally, by $\wp_u \in \mathcal{P}_{st}$ a path from s to t whose second node is a successor $u \in \text{Succ}(s)$ of s , and by $\wp'_u \in \mathcal{P}_{ut}$ the remainder of path \wp_u after the first step. The likelihood is then given by

$$\mathcal{L}(\beta | i) = \mathbb{P}(\nu_1 = i; \beta) = \sum_{\wp \in \mathcal{P}_{st}} \mathbb{P}_{st}(\wp) \mathbb{P}(\nu_1 = i | \rho = \wp) \quad (\text{A.11})$$

$$= \frac{1}{\mathcal{Z}_{st}} \sum_{\wp \in \mathcal{P}_{st}} \tilde{w}(\wp) \mathbf{P}(v_1 = i \mid \rho = \wp) \quad (\text{A.12})$$

$$= \frac{1}{\mathcal{Z}_{st}} \sum_{u \in \text{Succ}(s)} \sum_{\substack{\wp_u \in \mathcal{P}_{st} \\ \wp_u(1)=u}} \tilde{w}(\wp_u) \mathbf{P}(v_1 = i \mid \rho = \wp_u) \quad (\text{A.13})$$

$$= \frac{1}{\mathcal{Z}_{st}} \sum_{u \in \text{Succ}(s)} w_{su} \sum_{\wp'_u \in \mathcal{P}_{ut}} \tilde{w}(\wp'_u) \mathbf{P}(v_0 = i \mid \rho = \wp'_u) \quad (\text{A.14})$$

$$= \frac{1}{\mathcal{Z}_{st}} \sum_{u \in \text{Succ}(s)} w_{su} \mathbf{P}(v_0 = i) \mathcal{Z}_{ut} \quad (\text{A.15})$$

$$= \frac{1}{\mathcal{Z}_{st}} \sum_{u \in \text{Succ}(s)} w_{su} [\mathbf{L}_i]_{u,n+t}, \quad (\text{A.16})$$

where

$$\mathbf{P}(v_0 = i) \mathcal{Z}_{ut} = \sum_{\wp'_u \in \mathcal{P}_{ut}} \tilde{w}(\wp'_u) \mathbf{P}(v_0 = i \mid \rho = \wp'_u) \quad (\text{A.17})$$

derives from

$$\mathbf{P}(v_0 = i) = \sum_{\wp'_u \in \mathcal{P}_{ut}} \mathbf{P}(\rho = \wp'_u) \mathbf{P}(v_0 = i \mid \rho = \wp'_u) \quad (\text{A.18})$$

when considering paths from u to t , instead of from s to t . This is the desired result, as expressed in Equation (3.25). \square

B. Proof of Theorem 6 in the case of two observed edges

Here, we describe a more rigorous derivation, compared to the heuristic justification of Section 3.4, for the computation of the two-edge likelihood, that is, the likelihood of observing two edges $e_1 = (i_1, j_1)$, $e_2 = (i_2, j_2)$ along a path \wp . The derivation of the likelihood for the case of an arbitrary number of observed edges, as expressed in Equation (3.29) could be derived in a similar way. However, here we deal only with the two-edge case for conciseness and clarity.

We begin by rewriting the form presented in Equation (3.35) for the likelihood, with $M = 2$:

$$\mathcal{L}(\beta \mid (e_1, e_2)) = \frac{1}{\mathcal{Z}_{st}} \sum_{k=2}^{\infty} \frac{1}{\binom{k}{2}} \sum_{\wp_k \in \mathcal{P}_{st}^{(k)}} \tilde{w}(\wp_k) n_{e_1 \rightsquigarrow e_2}(\wp_k), \quad (\text{B.1})$$

where now $n_{e_1 \rightsquigarrow e_2}(\wp_k)$ denotes the number of times that edges e_1 and e_2 appear on \wp_k in that order, counting all occurrences of e_1 and e_2 as separate. For example, if $\wp = (1, 2, 3, 2, 3, 4)$, then $n_{(1,2) \rightsquigarrow (2,3)}(\wp) = 2$, as the edge $(2, 3)$ appears twice on \wp after the appearance of the edge $(1, 2)$.

Now, $n_{e_1 \rightsquigarrow e_2}$ can be calculated by iterating over all edges of the path, checking if the edge corresponds to e_1 , and then computing the number of times e_2 appears on the path after that. Summing the occurrences

of e_2 after each occurrence of e_1 then gives $n_{e_1 \rightsquigarrow e_2}$. Formally, for any $\wp_k \in \mathcal{P}_{st}^{(k)}$ with $k \geq 2$, where $\wp_k = (\wp(0), \wp(1), \dots, \wp(k))$, we have

$$n_{e_1 \rightsquigarrow e_2}(\wp_k) = \sum_{l=0}^{k-2} [\wp_k(l, l+1) = e_1] n_{e_2}(\wp_k(l+1 : k)), \quad (\text{B.2})$$

where $\wp_k(l_1 : l_2) = (\wp(l_1), \dots, \wp(l_2))$ denotes the subpath of \wp_k from the l_1 th node to the l_2 th node, with $0 \leq l_1 < l_2 \leq k$, and the brackets $[]$ are the Iverson brackets [58], that is, 1 if the statement within is true and 0 otherwise.

Inserting Equation (B.2) into the sum appearing in Equation (B.1), we can write the sum in matrix form, for path length $k \geq 2$ for path length $k = 2$ (using Equations (1.15), (3.7) and (3.13)):

$$\begin{aligned} & \sum_{\wp_k \in \mathcal{P}_{st}^{(k)}} \tilde{w}(\wp_k) n_{e_1 \rightsquigarrow e_2}(\wp_k) \\ &= \sum_{\wp_k \in \mathcal{P}_{st}^{(k)}} \sum_{l=0}^{k-2} \tilde{w}(\wp_k(0 : l)) [\wp_k(l, l+1) = e_1] \\ & \quad \times w_{i_1 j_1} \tilde{w}(\wp_k(l+1 : k)) n_{e_2}(\wp_k(l+1 : k)) \\ &= \sum_{l=0}^{k-2} \sum_{\wp_k \in \mathcal{P}_{st}^{(k)}} \tilde{w}(\wp_k(0 : l)) [\wp_k(l) = i_1] \\ & \quad \times w_{i_1 j_1} [\wp_k(l+1) = j_1] \left(-\frac{1}{\beta} \frac{\partial}{\partial c_{i_2 j_2}} \tilde{w}(\wp_k(l+1 : k)) \right) \\ &= -\frac{1}{\beta} \sum_{l=0}^{k-2} \mathbf{e}_s^T \tilde{\mathbf{W}}^l \mathbf{e}_{i_1} w_{i_1 j_1} \mathbf{e}_{j_1}^T \left(\frac{\partial \tilde{\mathbf{W}}^{k-l-1}}{\partial c_{i_2 j_2}} \right) \mathbf{e}_t \\ &= \frac{1}{\beta^2} \sum_{l=0}^{k-2} \mathbf{e}_s^T \tilde{\mathbf{W}}^l \left(\frac{\partial \tilde{\mathbf{W}}}{\partial c_{i_1 j_1}} \right) \left(\frac{\partial \tilde{\mathbf{W}}^{k-l-1}}{\partial c_{i_2 j_2}} \right) \mathbf{e}_t. \end{aligned} \quad (\text{B.3})$$

But in fact, the matrix appearing above,

$$\mathbf{S}_{e_1 \rightsquigarrow e_2}^{(k)} = \frac{1}{\beta^2} \sum_{l=0}^{k-2} \tilde{\mathbf{W}}^l \left(\frac{\partial \tilde{\mathbf{W}}}{\partial c_{i_1 j_1}} \right) \left(\frac{\partial \tilde{\mathbf{W}}^{k-l-1}}{\partial c_{i_2 j_2}} \right) \quad (\text{B.4})$$

can be computed, for any $k \geq 2$, as the **(1, 3)**-block of the k th power of matrix

$$\mathbf{Q}_{e_1 \rightsquigarrow e_2} = \begin{bmatrix} \tilde{\mathbf{W}} & \mathbf{W}_{i_1 j_1} & \mathbf{0} \\ \mathbf{0} & \tilde{\mathbf{W}} & \mathbf{W}_{i_2 j_2} \\ \mathbf{0} & \mathbf{0} & \tilde{\mathbf{W}} \end{bmatrix}, \quad (\text{B.5})$$

which is the two-edge version of the matrix $\mathbf{Q}_{\tilde{e}}$ presented in Equation (3.31). For this, note that we can write

$$\mathbf{Q}_{e_1 \rightsquigarrow e_2} = \begin{bmatrix} \bar{\mathbf{W}} & \bar{\mathbf{W}}_{i_1 j_1} & \mathbf{0} \\ \mathbf{0} & \bar{\mathbf{W}} & \bar{\mathbf{W}}_{i_2 j_2} \\ \mathbf{0} & \mathbf{0} & \bar{\mathbf{W}} \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{W}} & -\frac{1}{\beta} \partial_{c_{i_1 j_1}} \bar{\mathbf{W}} & \mathbf{0} \\ \mathbf{0} & \bar{\mathbf{W}} & -\frac{1}{\beta} \partial_{c_{i_2 j_2}} \bar{\mathbf{W}} \\ \mathbf{0} & \mathbf{0} & \bar{\mathbf{W}} \end{bmatrix}. \quad (\text{B.6})$$

It can then be shown (although, again, omitted for brevity) by induction, that, for all $k \geq 2$,

$$\mathbf{Q}_{e_1 \rightsquigarrow e_2}^k = \begin{bmatrix} \bar{\mathbf{W}}^k & -\frac{1}{\beta} \partial_{c_{i_1 j_1}} \bar{\mathbf{W}}^k & \frac{1}{\beta^2} \mathbf{S}_{e_1 \rightsquigarrow e_2}^{(k)} \\ \mathbf{0} & \bar{\mathbf{W}}^k & -\frac{1}{\beta} \partial_{c_{i_2 j_2}} \bar{\mathbf{W}}^k \\ \mathbf{0} & \mathbf{0} & \bar{\mathbf{W}}^k \end{bmatrix}. \quad (\text{B.7})$$

Using the above, we can finally write the two-edge likelihood, continuing from Equation (B.1) as

$$\mathcal{L}(\beta \mid (e_1, e_2)) = \frac{1}{\mathcal{Z}_{st}} \mathbf{e}_s^\top \sum_{k=2}^{\infty} \frac{1}{\binom{k}{2}} \mathbf{Q}_{e_1 \rightsquigarrow e_2}^k \mathbf{e}_{2n+t} = \frac{[\mathbf{L}_{e_1 \rightsquigarrow e_2}]_{s, 2n+t}}{\mathcal{Z}_{st}}, \quad (\text{B.8})$$

where

$$\mathbf{L}_{e_1 \rightsquigarrow e_2} = \sum_{k=2}^{\infty} \frac{1}{\binom{k}{2}} \mathbf{Q}_{e_1 \rightsquigarrow e_2}^k \quad (\text{B.9})$$

is the two-edge version of $\mathbf{L}_{\tilde{e}}$ defined in Equation (3.30).

As already mentioned, the proof of Equation (3.29) for the likelihood for an arbitrary number M of observed edges can be derived similarly to the process presented here. However, the derivation becomes overly tedious and messy when for an arbitrary M and is left out of the scope of this work.

C. Estimation of edge affinities as step selection probabilities

For the landscape graph used in Section 4, we estimated the edge affinities a_{ij} based on the data and general approach described in [19] from GPS data for more than 200 wild mountain reindeer from 7 of the largest wild reindeer management areas (including Austhei). As a slight difference, in [19] the affinities were based on estimating a Step Selection Function without an intercept, and therefore the model yielded values proportional, but not equal, to the probability of selecting a step. For the affinities a_{ij} in this paper, we used the same models, but we refitted them using the method developed in [59] to estimate actual probabilities of selection, using Step Selection Probability Functions (SSPF) [56, 60]. Using the `ResourceSelection` package [61] for R [62], we maximized the following log-likelihood (see [56] for details):

$$\mathcal{L}(\boldsymbol{\beta}; \mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m) = \sum_{k=1}^K [(\log \pi(\mathbf{x}_k; \boldsymbol{\beta}) - \log P_k(\boldsymbol{\beta}))] \quad (\text{C.1})$$

TABLE C1 *Fitted coefficients for the summer step selection model of reindeer. ‘Max.’ denotes the maximum value along the step, ‘prop.’ the proportion of the land cover class along the step, and ‘dens.’ is the road density, which is the length of road within a 5 km radius. The methods are detailed in [19], our only extension was to fit the models using a Step Selection Probability Function (see main text for further details).*

	Estimate	Std. error	z-value	Pr(> z)
Intercept	−3.79	0.79	− 4.77	<0.001
Step length	-1.24×10^{-3}	2.51×10^{-5}	−49.53	< 0.001
(max. slope) ²	-1.33×10^{-3}	4.79×10^{-5}	−27.81	< 0.001
Max. solar radiation	0.30	0.02	17.77	< 0.001
Max. trail dens.	−0.19	0.03	− 6.34	< 0.001
Max. road dens.	0.46	0.27	1.70	0.09
Crossing road	−1.19	0.37	−3.20	< 0.001
Prop. bog	−0.31	0.41	−0.76	0.45
Prop. non-forage	0.20	0.16	1.21	0.22
Prop. forage	0.88	0.14	6.37	< 0.001
Prop. lakes	−1.40	0.33	−4.21	< 0.001
Prop. reservoirs	−3.97	0.84	−4.70	< 0.001

where \mathbf{x}_k is the vector of M covariates associated to an observed step k between two consecutive locations, i and j , K is the number of observed steps in the data set and π the probability of selection. Furthermore,

$$P_k(\boldsymbol{\beta}) = \int \pi(\mathbf{x}; \boldsymbol{\beta}) f_k(\mathbf{x}; s_k) d\mathbf{x}, \quad (\text{C.2})$$

where $f_k(\mathbf{x}; s_k)$ denotes the distribution of resources available for step s_k . We defined the area within 2 km from the start location i of each step s_k as available. We used the Akaike Information Criterion for model selection (see [19] for details). Table C1 shows the parameter estimates that maximize the likelihood in Equation (C.1).

We predicted the affinities a_{ij} as the probability of a step between adjacent pixels i and j using the coefficients from the SSPF:

$$a_{ij} = \frac{\exp(\boldsymbol{\beta}_{\text{SSPF}}^T \mathbf{x})}{1 + \exp(\boldsymbol{\beta}_{\text{SSPF}}^T \mathbf{x})}, \quad (\text{C.3})$$

where $\boldsymbol{\beta}_{\text{SSPF}}$ is a row vector with M elements corresponding to the coefficients from the SSPF (see Table C1), and \mathbf{x} is a column vector with M elements describing the environmental characteristics of the transition (first element is the intercept, and equals 1). Thus, a_{ij} is the probability of selection of step i -to- j (instead of staying put) based on the vector of covariates (e.g. geographic distance between i and j , road crossing, proportion of each land cover) characterizing this transition.

REFERENCES

1. BARABASI, A. L. (2016) *Network Science*. Cambridge University Press, Cambridge, MA, USA.
2. ESTRADA, E. (2012) *The Structure of Complex Networks*. Oxford University Press, Oxford, UK.
3. FOUSS, F., SAERENS, M. & SHIMBO, M. (2016) *Algorithms and Models for Network Data and Link Analysis*. Cambridge University Press, Cambridge, MA, USA.
4. KOLACZYK, E. (2009) *Statistical Analysis of Network Data: Methods and Models*. Springer, New York City, NY, USA.
5. NEWMAN, M. (2010) *Networks: An Introduction*. Oxford University Press, Oxford, UK.
6. WASSERMAN, S. & FAUST, K. (1994) *Social Network Analysis: Methods and Applications*. Cambridge University Press, Cambridge, MA, USA.
7. CHANDRA, A. K., RAGHAVAN, P., RUZZO, W. L., SMOLENSKY, R. & TIWARI, P. (1989) The electrical resistance of a graph captures its commute and cover times. In Proceedings of the twenty-first annual ACM symposium on Theory of computing (STOC '89). Association for Computing Machinery, New York, NY, USA, 574–586. DOI:<https://doi.org/10.1145/73007.73062>.
8. KIVIMÄKI, I., SHIMBO, M. & SAERENS, M. (2014) Developments in the theory of randomized shortest paths with a comparison of graph node distances. *Physica A*, **393**, 600–616.
9. KLEIN, D. J. & RANDIC, M. (1993) Resistance distance. *J. Math. Chem.*, **12**, 81–95.
10. VON LUXBURG, U., RADL, A. & HEIN, M. (2010) Getting lost in space: Large sample analysis of the resistance distance. In Lafferty, J. D., Williams, C. K. I., Shawe-Taylor, J., Zemel, R. S. & Culotta, A., editors, *Advances in Neural Information Processing Systems 23*, pages 2622–2630. Curran Associates, Inc.
11. VON LUXBURG, U., RADL, A. & HEIN, M. (2014) Hitting and commute times in large random neighborhood graphs. *J. Mach. Learn. Res.*, **15**, 1751–1798.
12. NGUYEN, C. H. & MAMITSUKA, H. (2016) New Resistance Distances with Global Information on Large Graphs. In Gretton, A. & Robert, C. C., editors, Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, volume 51 of Proceedings of Machine Learning Research, pages 639–647, Cadiz, Spain. PMLR.
13. ODOR, G. (2017) Global information loss and criticality in resistance matrices. Master's Thesis, Central European University.
14. KIVIMÄKI, I., LEBICHOT, B., SARAMÄKI, J. & SAERENS, M. (2016) Two betweenness centrality measures based on randomized shortest paths. *Sci. Rep.*, **6**, 19668.
15. SAERENS, M., ACHBANY, Y., FOUSS, F. & YEN, L. (2009) Randomized shortest-path problems: two related models. *Neural Comput.*, **21**, 2363–2404.
16. YEN, L., MANTRACH, A., SHIMBO, M. & SAERENS, M. (2008) A family of dissimilarity measures between nodes generalizing both the shortest-path and the commute-time distances. *Proceedings of the 14th SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2008)*. pp. 785–793. New York, NY, USA. Association for Computing Machinery.
17. KIVIMÄKI, I. (2018) Distances, centralities and model estimation methods based on randomized shortest paths for network data analysis. PhD Thesis, UCLouvain - SST/ICTM/INGI - Pôle en ingénierie informatique SST/ICTM - Institute of Information and Communication Technologies, Electronics and Applied Mathematics - Ecole Polytechnique de Louvain.
18. GARCIA-DIEZ, S., VANDENBUSSCHE, E. & SAERENS, M. (2011) A continuous-state version of discrete randomized shortest-paths, with application to path planning. *2011 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*. pp. 6570–6577.
19. PANZACCHI, M., VAN MOORTER, B., STRAND, O., SAERENS, M., KIVIMÄKI, I., ST CLAIR, C. C., HERFINDAL, I. & BOITANI, L. (2016) Predicting the continuum between corridors and barriers to animal movements using step selection functions and randomized shortest paths. *J. Anim. Ecol.*, **85**, 32–42.
20. VAN ETTEN, J. (2017) R Package gdistance: Distances and Routes on Geographical Grids. *Journal of Statistical Software, Articles*, **76**, 1–21.

21. AKAMATSU, T. (1996) Cyclic flows, Markov process and stochastic traffic assignment. *Transport. Res. B*, **30**, 369–386.
22. FRANÇOISSE, K., KIVIMÄKI, I., MANTRACH, A., ROSSI, F. & SAERENS, M. (2017) A bag-of-paths framework for network data analysis. *Neural Netw.*, **90**, 90–111.
23. Golnari, G., Zhang, Z.-L. & Boley, D. (2019) Markov fundamental tensor and its applications to network analysis. *Linear Algebra and its Applications*, **564**, 126–158.
24. KAPUR, J. N. & KESAVAN, H. K. (1992) *Entropy Optimization Principles with Applications*. Academic Press, San Diego, CA, USA.
25. GRINSTEAD, C. & SNELL, J. L. (1997) *Introduction to Probability*, 2nd ed. The Mathematical Association of America, Providence, RI, USA.
26. MEYER, C. D. (2000) *Matrix Analysis and Applied Linear Algebra*, vol. 71. SIAM, Philadelphia, PA, USA.
27. FREEMAN, L. (1978) Centrality in social networks conceptual clarification. *Soc. Netw.*, **1**, 215 – 239.
28. MANTRACH, A., YEN, L., CALLUT, J., FRANCOISSE, K., SHIMBO, M. & SAERENS, M. (2010) The sum-over-paths covariance kernel: a novel covariance measure between nodes of a directed graph. *IEEE Trans. Pattern Anal. Mach. Intell.*, **32**, 1112–1126.
29. LEBICHOT, B., KIVIMÄKI, I., FRANÇOISSE, K. & SAERENS, M. (2014) Semisupervised classification through the bag-of-paths group betweenness. *IEEE Trans. Neural Netw. Learn. Syst.*, **25**, 1173–1186.
30. BEN-AKIVA, M. & BIERLAIRE, M. (1999) Discrete choice methods and their applications to short term travel decisions. In Hall, R. W., *Handbook of Transportation Science*, pp.5–33. Springer US, Boston, MA.
31. BEN-AKIVA, M. E., LERMAN, S. R. (1985) *Discrete Choice Analysis: Theory and Application to Travel Demand*, vol. 9. MIT Press, Cambridge, MA, USA.
32. DIAL, R. B. (1971) A probabilistic multipath traffic assignment model which obviates path enumeration. *Transport. Res.*, **5**, 83–111.
33. PRASHKER, J. N. & BEKHOR, S. (2004) Route choice models used in the stochastic user equilibrium problem: a review. *Transport Rev.*, **24**, 437–463.
34. OYAMA, Y. & HATO, E. (2017) A discounted recursive logit model for dynamic gridlock network analysis. *Transport. Res. C*, **85**, 509–527.
35. OYAMA, Y. & HATO, E. (2019) Prism-based path set restriction for solving Markovian traffic assignment problem. *Transport. Res. B*, **122**, 528–546.
36. BEN-AKIVA, M. E. (1973) Structure of passenger travel demand models. PhD Thesis, Massachusetts Institute of Technology.
37. MAI, T., FOSGERAU, M. & FREJINGER, E. (2015) A nested recursive logit model for route choice analysis. *Transport. Res. B*, **75**, 100–112.
38. MATĚJKA, F. & MCKAY, A. (2015) Rational inattention to discrete choices: a new foundation for the multinomial logit model. *Am. Econ. Rev.*, **105**, 272–298.
39. FISK, C. (1977) Note on the maximum likelihood calibration on Dial’s assignment method. *Transport. Res.*, **11**, 67–68.
40. ROBILLARD, P. (1974) Calibration of Dial’s assignment method. *Transport. Sci.*, **8**, 117–125.
41. ANAS, A. (1988) Statistical properties of mathematical programming models of stochastic network equilibrium. *J. Region. Sci.*, **28**, 511–530.
42. FOSGERAU, M., FREJINGER, E. & KARLSTROM, A. (2013) A link based network route choice model with unrestricted choice set. *Transport. Res. B*, **56**, 70–80.
43. MAI, T. (2016) A method of integrating correlation structures for a generalized recursive route choice model. *Transport. Res.*, **93**, 146–161.
44. OYAMA, Y. & HATO, E. (2018) Link-based measurement model to estimate route choice parameters in urban pedestrian networks. *Transport. Res. C*, **93**, 62–78.
45. CRAIG, B. A. & SENDI, P. P. (2002) Estimation of the transition matrix of a discrete-time Markov chain. *Health Econ.*, **11**, 33–42.
46. METZNER, P., DITTMER, E., JAHNKE, T. & SCHÜTTE, C. (2007) Generator estimation of Markov jump processes. *J. Comput. Phys.*, **227**, 353–375.

47. SHELTON, C. R. & CIARDO, G. (2014) Tutorial on structured continuous-time Markov processes. *J. Artif. Intell. Res.*, **51**, 725–778.
48. QIN, F., AUERBACH, A. & SACHS, F. (1997) Maximum likelihood estimation of aggregated Markov processes. *Proc. R. Soc. Lond. B*, **264**, 375–383.
49. MEYER, C. D. (1989) Stochastic complementation, uncoupling Markov chains, and the theory of nearly reducible systems. *SIAM Rev.*, **31**, 240–272.
50. BLADT, M. & SØRENSEN, M. (2005) Statistical inference for discretely observed Markov jump processes. *J. R. Stat. Soc. B*, **67**, 395–410.
51. LANCICHINETTI, A. & FORTUNATO, S. (2009) Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities. *Phys. Rev. E*, **80**, 016118.
52. RYU, S., CHEN, A., XU, X. & CHOI, K. (2014) A dual approach for solving the combined distribution and assignment problem with link capacity constraints. *Netw. Spatial Econ.*, **14**, 245–270.
53. GUEX, G., KIVIMÄKI, I. & SAERENS, M. (2019) Randomized optimal transport on a graph: framework and new distance measures. *Netw. Sci.*, **7**, 88–122.
54. HIGHAM, N. J. (2008) *Functions of Matrices: Theory and Computation*. SIAM, Philadelphia, PA, USA.
55. AL-MOHY, A. H., HIGHAM, N. J. & RELTON, S. D. (2013) Computing the Fréchet derivative of the matrix logarithm and estimating the condition number. *SIAM J. Sci. Comput.*, **35**, C394–C410.
56. LELE, S. R. & KEIM, J. L. (2006) Weighted distributions and estimation of resource selection probability functions. *Ecology*, **87**, 3021–3028.
57. PANZACCHI, M., VAN MOORTER, B. & STRAND, O. (2013) A road in the middle of one of the last wild reindeer migration routes in Norway: crossing behaviour and threats to conservation. *Rangifer*, **33**, 15–26.
58. KNUTH, D. E. (1992) Two notes on notation. *Am. Math. Monthly*, **99**, 403–422.
59. LELE, S. R. (2009) A new method for estimation of resource selection probability function. *J. Wildlife Manage.*, **73**, 122–127.
60. FORESTER, J. D., IM, H. K. & RATHOUZ, P. J. (2009) Accounting for animal movement in estimation of resource selection functions: sampling and data analysis. *Ecology*, **90**, 3554–3565.
61. LELE, S. R., KEIM, J. L. & SOLYMOS, P. (2017) *ResourceSelection: Resource Selection (Probability) Functions for Use-Availability Data*. R package version 0.3-2.
62. Core Team (2015) *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing.