Detecting effects of spatial memory and dynamic information on animal movement decisions

Ulrike E. Schlägel, Mark A. Lewis.

Appendix A Supplemental methods

A.1 Simulation of landscapes

We modelled the continuous valued environmental covariate as a Gaussian random field (Haran, 2011). A Gaussian random field is a multivariate Gaussian random variable, indexed by space. In our case, the random variable is the resource r_1 for each location in the spatial domain, $\{r_1(\boldsymbol{x})\}_{\boldsymbol{x}\in\Omega}$. The covariance between resource values at any two locations $\boldsymbol{x}$ and $\boldsymbol{y}$ is a function of the distance between the locations, so that values of nearby locations are stronger correlated than values of locations that are far apart. We chose the exponential form for the covariance function, $\operatorname{cov}\big(r_1(\boldsymbol{x}), r_1(\boldsymbol{y})\big) = \exp(-\frac{\|\boldsymbol{x}-\boldsymbol{y}\|}{\sigma})$, where σ determines the rate at which locations cease to be correlated. We varied σ among different landscapes. To simulate such landscapes, we used the R package ‘RandomFields’ (Schlather *et al.*, 2013).

To generate correlated landscapes of binary variables we use the method and C code provided by Hiebeler (2000). Each landscape is represented by two quantities: p_0, the overall proportion of type 0 cells, and q_{00}, the probability that a neighbour of a type 0 cell is also of type 0. If q_{00} is high, the landscape is strongly clustered, and vice versa. For our landscapes, we varied both p_0, and their degree of clustering, q_{00}.

The five landscape pairs we used for our simulations are depicted in Fig. 1

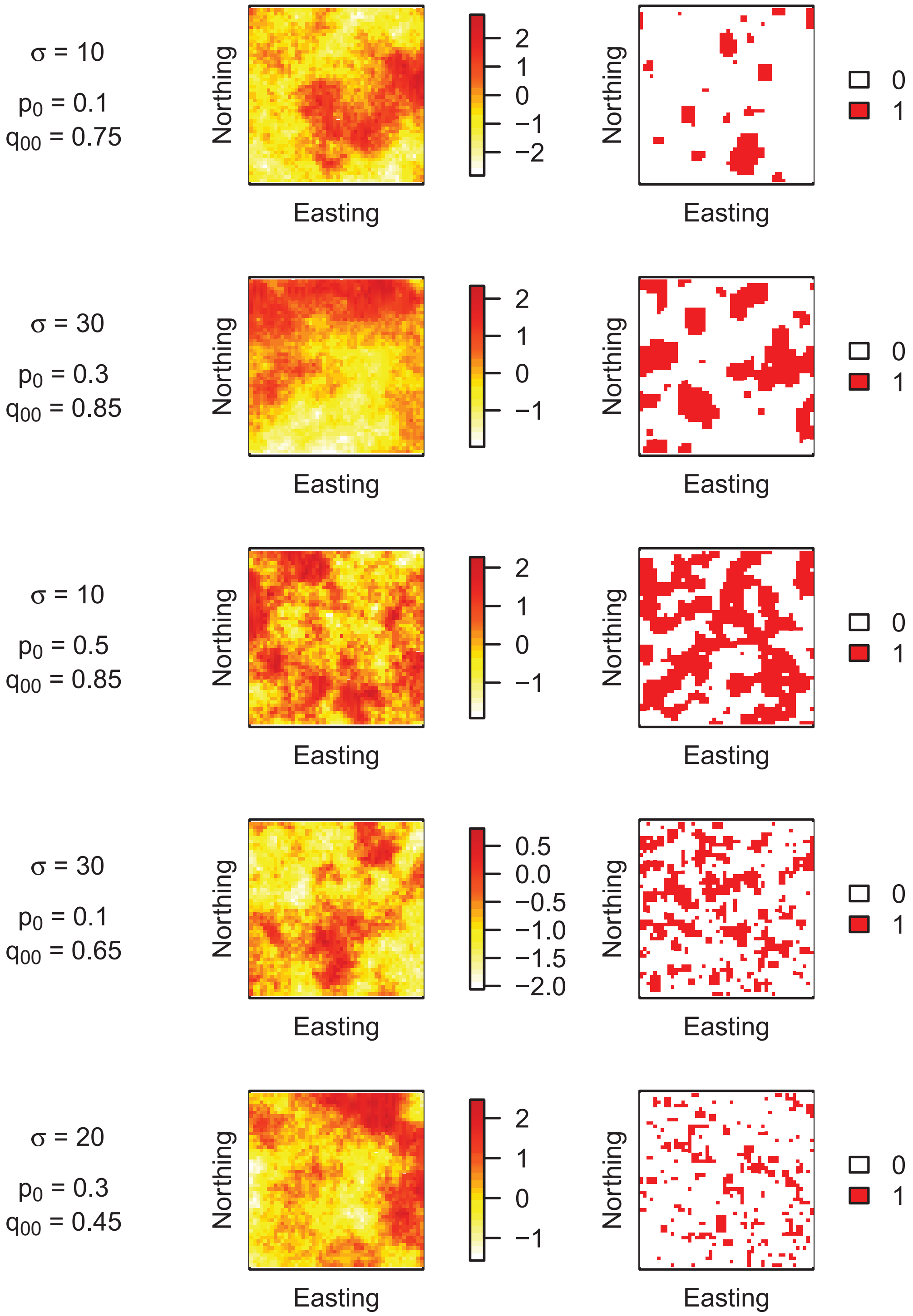


Fig. 1. Pairs of simulated landscapes (row-wise). The left side shows the continuous valued resource r_1, the left side shows the binary variable r_2. Parameter values used to simulate the landscapes are given for every landscape pair.

A.2 Data cloning and MCMC in simulation analysis

For all model fits, we used data cloning. Data cloning uses Markov Chain Monte Carlo (MCMC) methods, which calculate a posterior distribution for the model parameters, given the data. This technique is usually employed in Bayesian statistical inference, however, the resulting parameter estimates from data cloning approximate the corresponding maximum likelihood estimates (MLE). This is achieved by applying the Bayesian framework to K copies of the data, which are referred to as clones. Alternatively, the procedure can be viewed as a series of Bayesian updates applied to the same data, each time using the posterior distribution from the previous update as new prior distribution (Robert, 1993). After a movement trajectory has been cloned K times, Bayesian parameter estimation, here MCMC, is performed on this augmented data. The results of this procedure lead to parameter estimates in the more conventional style of frequentist inference, namely maximum likelihood estimates. However, an important factor to achieve this is a sufficiently large number of clones. If K is large enough, the posterior distribution for the parameters is approximately Normal with mean at the maximum likelihood estimate $\hat{\boldsymbol{\theta}}_{\mathrm{MLE}}$ of the original (i.e. uncloned) data and with variance $\frac{1}{K} I^{-1}(\hat{\boldsymbol{\theta}}_{\mathrm{MLE}})$, where I is the Fisher information of the original data (Lele *et al.*, 2010). This means that if we choose K large enough, the sample mean of the MCMC is approximately the MLE of the original uncloned trajectory and if we multiply the sample variance by K, we obtain an approximation of the inverse Fisher information (also termed 'information number'; Casella & Berger, 2001). The inverse of the Fisher information is the asymptotic variance of a maximum likelihood estimate and it can be used to calculate Wald-type confidence intervals.

In our analysis, we used $K = 15$ clones. To confirm that this number of clones was sufficient to obtain a good Normal approximation of the posterior distributions and good approximations of the maximum likelihood estimates, we performed test runs with the most complex combination model. We selected a combination trajectory both from the main data set and the supplemental data and iteratively fitted the combination model with increasing number of clones. For each run, we inspected the three diagnostic measures described in Lele *et al.* (2010) and Solymos (2010) (`lambda.max, ms.error, r.squared`). These diagnostics assess whether the Normal approximation of the posterior distributions and the approximation of the sample mean to the MLE are adequate, which is the case if the diagnostics converge to zero. We found that all three diagnostics converged to zero for our test fits, and that they were all close to zero (< 0.05) for $K = 15$.

For the MCMC, we used two parallel chains, each running for 7500 iterations, of which we discarded an initialization and burn-in period of 3500 iterations. To assess whether this was sufficient to obtain good mixing properties of the chains and convergence to the stationary distribution, we inspected the Markov chains visually and calculated the *potential scale reduction factor* $\hat{R}$ (Brooks & Gelman, 1998) for each parameter. Using these amounts of MCMC iterations, we obtained good mixing and convergence in matching model fits. In non-matching model fits (model and simulated trajectory mechanism did not match), mixing and convergence problems occurred. To ensure that these problems did not simply occur because we did not use sufficiently many MCMC iterations, we continued to run some of the non-mixing/non-converging MCMCs for up to 8000 additional iterations. In none of these cases we found that more iterations improved mixing or convergence.

For further analysis, we calculated data cloning estimability diagnostic for selected trajectories. This requires a series of model fits with increasing number of clones, for which we used the functions `dc.fit` and `dc.parfit` from the R data cloning package (Solymos, 2010). We chose number of clones $K = 1, 5, 10, 15$. Because of the high computational needs of the model fits, especially for the most complex combination model, we refrained from increasing the number of clones further. However, as additional test we also examined one- and two-dimensional slices of the corresponding likelihood functions and found that these tests always lead to the same conclusions; see section B.3. If in a model fit parameters are estimable, their variances should decrease with increasing number of clones. In particular, the largest eigenvalue of the posterior variance, `lambda.max`, should decrease with rate $\frac{1}{K}$ (Lele *et al.*, 2010; Solymos, 2010). Standardized by its value for the uncloned data, it should converge to zero as $\frac{1}{K}$.

A.3 Missed observations

In general, there are several approaches how to deal with missed observations in a trajectory statistically. The easiest case is when locations, or steps, are modelled as being independent from each other. However, for models that include autocorrelation, we cannot simply ignore the dependency structure. A possibility is to use some technique of interpolation. Alternatively, we can divide the trajectory into chunks of available data and condition the likelihood function on the first available observation in each chunk. In a correlated random walk, we need three consecutive locations to define one step probability. Therefore, missed locations effectively lead to even larger gaps in the likelihood function.

To avoid any loss of data, we can use the full likelihood based on the entire trajectory $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = (\boldsymbol{x}_{\text{observed}}, \boldsymbol{x}_{\text{missed}})$ and integrate over all missed observations,

$$L(\boldsymbol{\theta}|(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)) = L(\boldsymbol{\theta}|(\boldsymbol{x}_{\text{observed}}, \boldsymbol{x}_{\text{missed}})) = \int p(\boldsymbol{x}_{\text{observed}}, \boldsymbol{x}_{\text{miss}}|\boldsymbol{\theta})\, d\boldsymbol{x}_{\text{miss}}. \qquad \text{eqn 1}$$

This has the advantage that all original dependencies between locations can be preserved and no information is lost. Calculation of the possibly high-dimensional integral is problematic in common frequentist methods that require optimization of the likelihood function. However, MCMC techniques (and therefore data cloning) circumvent this problem and at the same time provide estimates for the missed variables.

Our model is formulated based entirely on locations (intermediate quantities such as step length and bearing are calculated within the model formulation), and therefore implementation of this method is, in principle, straightforward: in the MCMC, missed locations are treated as parameters and their step probabilities serve as priors. We used JAGS for model fitting, which was capable to perform this and to produce converging posterior distributions for missed locations. However, this came at the cost of very high computational needs (both memory requirements and computation time).

The memory model requires reconstruction of time since last visit $\boldsymbol{m}$. For a missed location at time t, we accordingly miss m_t. Because m_t is a function of $\boldsymbol{x}_t$ and $\boldsymbol{x}_{t-1}$, just as step length and bearing, we could estimate m_t within the model fitting procedure. However, m_t is a high-dimensional variable for each time step and due to computational restrictions we treat $\boldsymbol{m}$ as known covariate. Therefore, if the location $\boldsymbol{x}_t$ is missing, we do not update

m_t for this time step and set $m_t = m_{t-1}$. At the next time step, we update m_{t+1} via the usual formula, but based on m_{t-1} and $\boldsymbol{x}_{t-1}$. To account for the longer time, we increase the distance δ from the straight line path($\boldsymbol{x}_{t-1} \rightarrow \boldsymbol{x}_{t+1}$), in which locations are considered as visited. If more than one location is missed in a row, we proceed similar, starting to update time since last visit at the next available location.

To perform model selection for a completely observed trajectory, it is possible with our models to calculated the likelihood functions and thus BIC. With missed locations, this becomes computationally much more complex due to the integration. To avoid this, we can, as an approximation, instead use estimates of missed locations. Because we treat missed locations as parameters, we obtain posterior distributions and estimates for them. We can simply use these estimates to calculate the likelihood function. A more sophisticated method has been proposed by Ponciano *et al.* (2009). Their method circumvents the problem of integration and uses data cloning itself to obtain estimates of likelihood ratios, which can then be used for AIC or BIC.

Appendix B Supplemental results

B.1 Supplemental data

In the main text of the paper, we analyzed a simulated data set of 20 trajectories. This data set was generated using realistic parameter values. However, to test our method in even more scenarios, we generated two additional sets of 20 trajectories, which we refer to as 'data set 2' and 'data set 3' to separate them easier from the main data set. For these data, we used parameter sets in which we included values that we considered to be potentially more difficult to estimate from data.

For data set 2, we chose relatively small resource selection parameters α_{res} β_1, β_2 and very small interaction parameters γ_1, γ_2, which means that we simulated weak effects of the resources. The parameters α_{mem} and β_{mem} that regulate the influence of time since last visit were chosen so that returns to locations were possible again after short durations of absence. This means that the effect of time since last visit is relatively weak. The parameter values were

$$\begin{array}{lllll}
\text{parameter set 2:} & \kappa = 4 & \alpha_{\text{res}} = -0.2 & \beta_1 = 0.5 & \gamma_1 = 0.008 \\
 & \lambda = 0.9 & \alpha_{\text{mem}} = -3 & \beta_2 = 0.8 & \gamma_2 = 0.005 \\
 & \rho = 1.2 & \alpha_{\text{com}} = -3.2 & \beta_{\text{mem}} = 0.04 &
\end{array}$$

For data set 3, we set one of the interaction parameters in the combination model to zero, so that an interaction between resource values and time since last visit was only present for the binary variable r_2. To distinguish this data set further from the main set, we chose $\beta_1 < 0$, so that resource variable r_1 has an opposite effect compared to the other data sets. All other parameters are again chosen to be realistic, but different from previous values.

$$\begin{array}{lllll}
\text{parameter set 3:} & \kappa = 4.5 & \alpha_{\text{res}} = 0.8 & \beta_1 = -1.5 & \gamma_1 = 0 \\
 & \lambda = 1.3 & \alpha_{\text{mem}} = -5 & \beta_2 = 2.5 & \gamma_2 = 0.01 \\
 & \rho = 1.5 & \alpha_{\text{com}} = -5.8 & \beta_{\text{mem}} = 0.05 &
\end{array}$$

The simulated resource landscapes were the same as for the main data set. We performed the same analysis on the supplemental data as on the main data and we obtained 160 model fits.

B.2 Results

B.2.1 Results on supplemental data sets

Considering the 160 model fits, 82% had potential scale reduction factor $0.9 \leq \hat{R} \leq 1.1$ for all parameters, which means that for those model fits, MCMC runs for all parameters mixed well and converged. If convergence and mixing problems occurred, these were cases where parameters were inapplicable to the analyzed trajectory (Fig. 2).

In contrast to the main data set, there was one instance in the supplemental data where $\hat{R} > 1.1$ for several parameters in a matching model fit in data set 2. This was a combination trajectory fitted with the matching combination model. We continued to run the Markov chains for more iterations, however, the chains' behaviour remained the same. We therefore looked closer into this model fit, and ultimately ran an entirely new MCMC for this trajectory to calculate estimability diagnostics. In this second run, the parallel chains mixed and converged well and no estimability issues were found. For more details and discussion about this, see section B.3.

Model selection via BIC was able to correctly identify true underlying models for all but one trajectory (Fig. 2). For the fourth resource trajectory of data set 3, BIC was lowest for the null model, followed by the resource model. When we looked closer at the results of the fit with the resource model, we found that selection parameters α_{res} and β_1 had very large confidence intervals, and the estimate of the intercept α_{res} was high. High values of the intercept effectively result in a constant weighting function, thereby mimicking the null model. We discuss this model fit further in section B.3. Additionally, there was one matching combination model fit that did not mix properly in the first MCMC run (compare previous paragraph), and for which we therefore did not calculate BIC. This lead to the memory model being selected as best model for a combination trajectory. However, when we ran a new MCMC with the combination model, the chains converged and BIC was lowest, followed in order by the memory model ($\Delta\text{BIC} = 178$), the resource model ($\Delta\text{BIC} = 676$) and the null model ($\Delta\text{BIC} = 800$).

In the following, we used results from the second MCMC run for the fourth combination trajectory in data set 2.

For our hypothesis test, we obtained 139 estimates of selection parameters (we only considered estimates from convergent and well mixing Markov chains). Of these, 69 corresponded to true underlying effects. When we analyzed confidence intervals as to whether they excluded zero and thus suggested covariate effects, we obtained a Type I error rate of 0.01 (a trajectory was simulated without effect, but confidence intervals detect an effect) and a Type II error rate of 0.14 (a trajectory was simulated with an effect, which was not detected). When we pooled supplemental and main data, we obtained a Type I error rate of 0.04, which is close to the expected amount if we use 95% confidence intervals (which corresponds to a 5%-level hypothesis test). For the pooled data, the Type II error rate was 0.09. Hence, overall the hypothesis test gives expected results that include errors, while the model selection via BIC performs better and reliably identifies trajectories' true underlying mechanisms.

Most parameter estimates of matching model fits agreed well with true underlying values. As expected, 95% confidence intervals (n=230) included the true value 0.95% of the time. In data set 2, there was one resource trajectory, for which the estimate of α_{res} was far away from the true value ($\hat{\alpha}_{\text{res}} = -9.7$, true value was -0.2) and the standard error was very large (sd=21.7). We looked into this further and calculated likelihood slices and data cloning estimability diagnostics. From these, we concluded that there is an estimability problem for α_{res}, while the other parameters are well behaved; for details see section B.3. Therefore, we excluded this estimate of α_{res}. All remaining estimates in dats set 2 were balanced around and generally close to their true values (Fig. 3, panel (a)). In Fig. 3, panel (a), we plotted estimates for α_{res} and γ_2 separately, using the original unscaled values, because their standardized confidence intervals were larger than for the other parameters. Standardization is sensitive to the size of the standardization constant and may be problematic here, because the true values of α_{res} and γ_2 are small, and division by values close to zero results in large

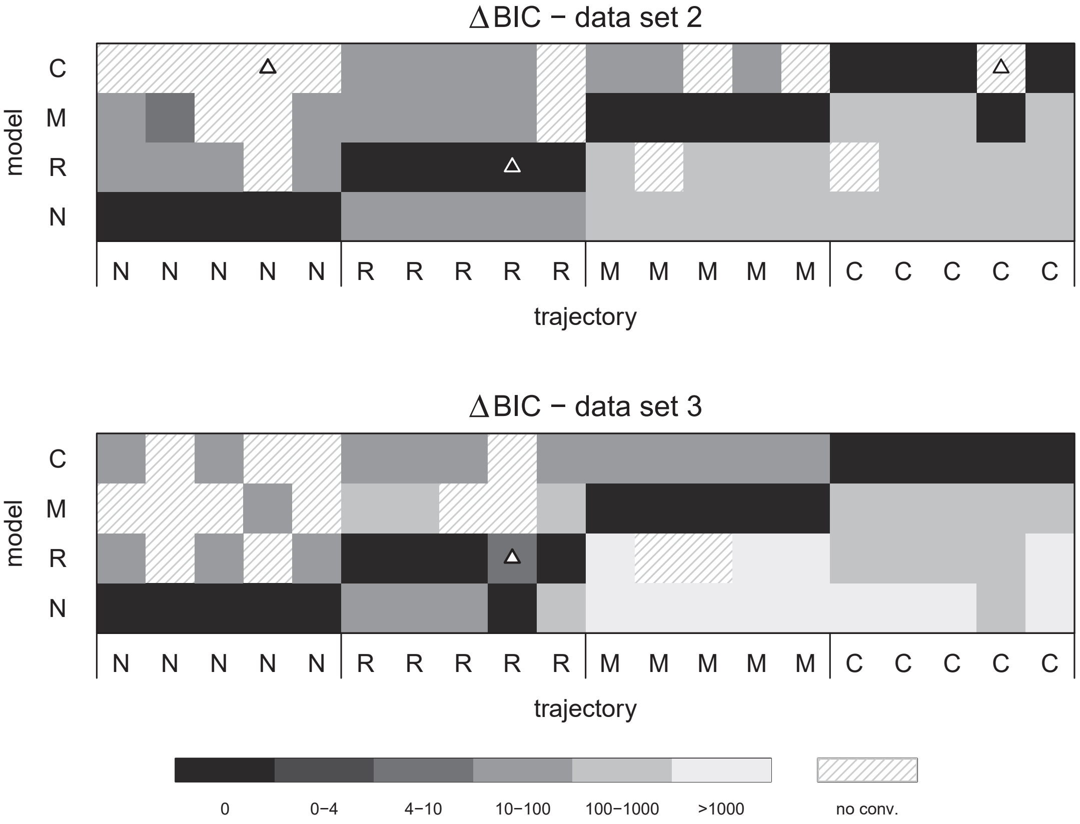


Fig. 2. Each column in the two subfigures shows model selection results for one simulated trajectory when it was fitted with the four candidate models (null, resource, memory, combination). For each trajectory, we calculated BIC values for the four fitted models, and the figure shows differences in BIC with respect to the minimal BIC value, i.e., the model with minimal BIC has $\Delta\text{BIC} = 0$. We excluded model fits with non-convergent MCMC. For coherence, the figures depict the results from the first MCMC run for each trajectory. Triangle indicate those trajectories for which we calculated estimability diagnostics.

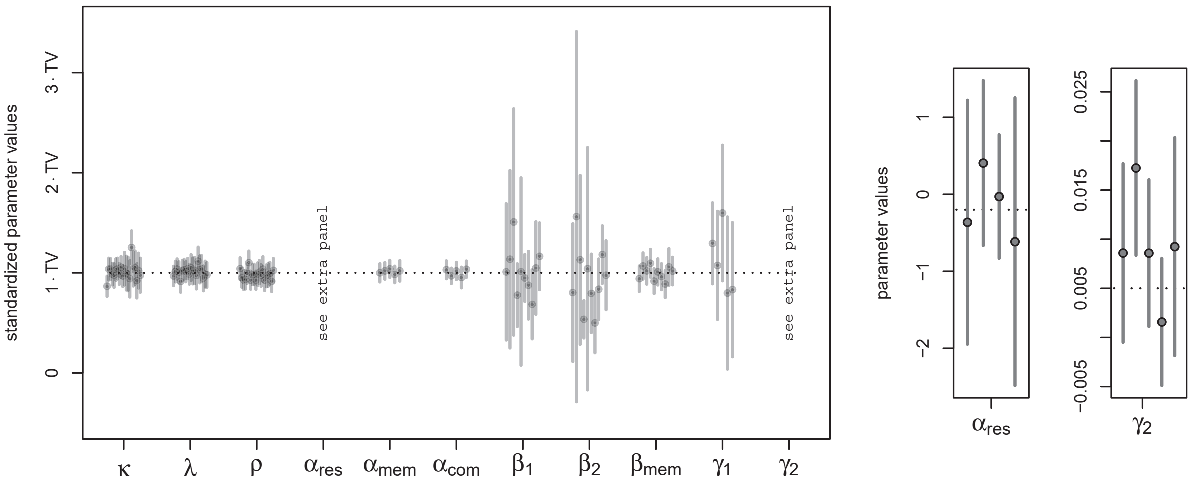


Fig. 3. Parameter estimates and their 95% confidence intervals for matching model fits. In large panels, both parameter estimates and Wald-type confidence intervals are scaled by the true parameter values (TV): $\kappa = 4$, $\lambda = 0.9$, $\rho = 1.2$, $\alpha_{\text{res}} = -0.2$, $\alpha_{\text{mem}} = -3$, $\alpha_{\text{com}} = -3.2$, $\beta_1 = 0.5$, $\beta_2 = 0.8$, $\beta_{\text{mem}} = 0.04$, $\gamma_1 = 0.008$, γ_2=0.005. Smaller panels have unscaled values. In all plots, dotted lines mark true values.

values. The unscaled results for α_{res} and γ_2 look reasonable (Fig. 3, smaller panels).

In data set 3, there were a few more estimates with large confidence intervals (Fig. 4, panel (b)), particularly for β_2 and γ_2, which we have therefore plotted in separate panels in their original scale. We suspect that these large confidence intervals are due to estimability problems; see also discussion in section B.3. We also plotted γ_1 separately, because its true value was zero and therefore could not be standardized. All estimates of γ_1 are close to zero and all confidence intervals overlap zero. Therefore, the model was able to correctly identify the lacking effect of the interaction parameter.

B.2.2 Missed observations

For the combination trajectory with missed locations, we performed a matching model fit. We compared parameter estimates and their 95% confidence intervals for the trajectory with missed locations and the corresponding complete trajectory. Parameter estimates for the combination trajectory with missed locations agreed well with true values and were similar to results for the complete trajectory (Fig. 5). Estimates of selection parameters tended to be slightly lower for the incomplete trajectory, but standardized values never deviated more than 0.15. Parameters κ, λ of the movement kernel describing step lengths (shape and scale of Weibull distribution) are slightly higher for the incomplete trajectory, resulting in a mean step length of 5.29 compared to 5.0 for the complete trajectory.

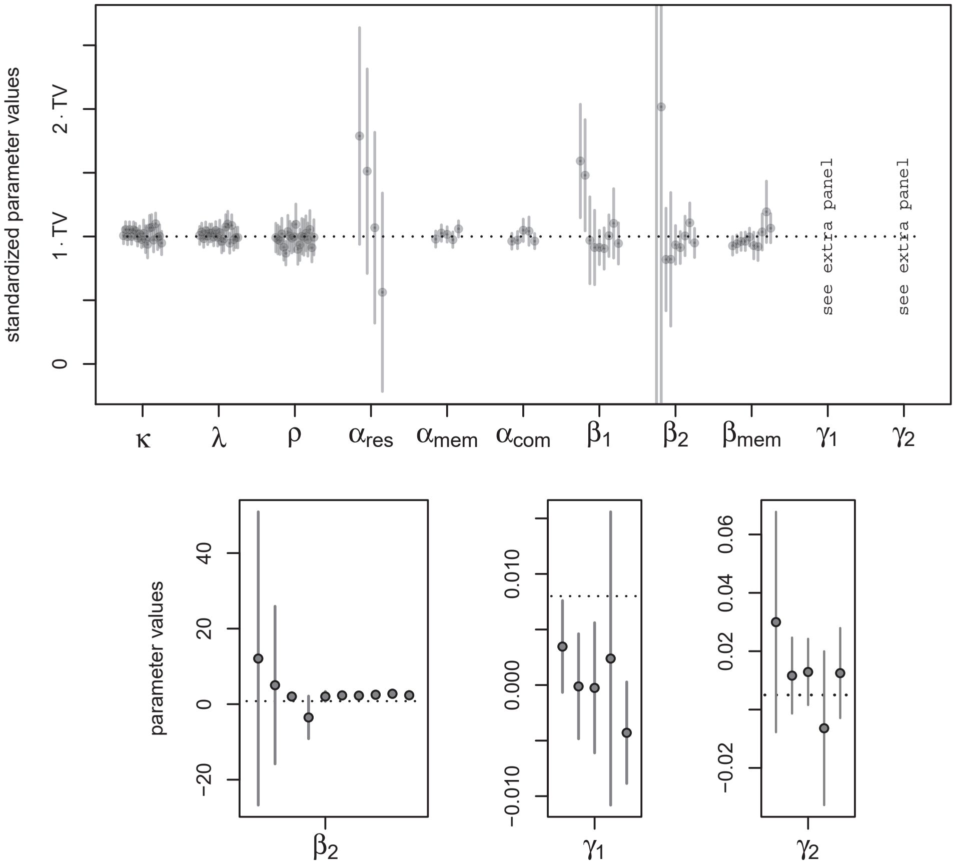


Fig. 4. Parameter estimates and their 95% confidence intervals for matching model fits. In large panels, both parameter estimates and Wald-type confidence intervals are scaled by the true parameter values (TV): $\kappa = 4.5$, $\lambda = 1.3$, $\rho = 1.5$, $\alpha_{\text{res}} = 0.8$, $\alpha_{\text{mem}} = -5$, $\alpha_{\text{com}} = -5.8$, $\beta_1 = -1.5$, $\beta_2 = 2.5$, $\beta_{\text{mem}} = 0.05$, $\gamma_1 = 0$, γ_2=0.01. Smaller panels have unscaled values. In all plots, dotted lines mark true values. Estimates from the matching resource model fit in data set 3 that was not selected as best model are excluded.

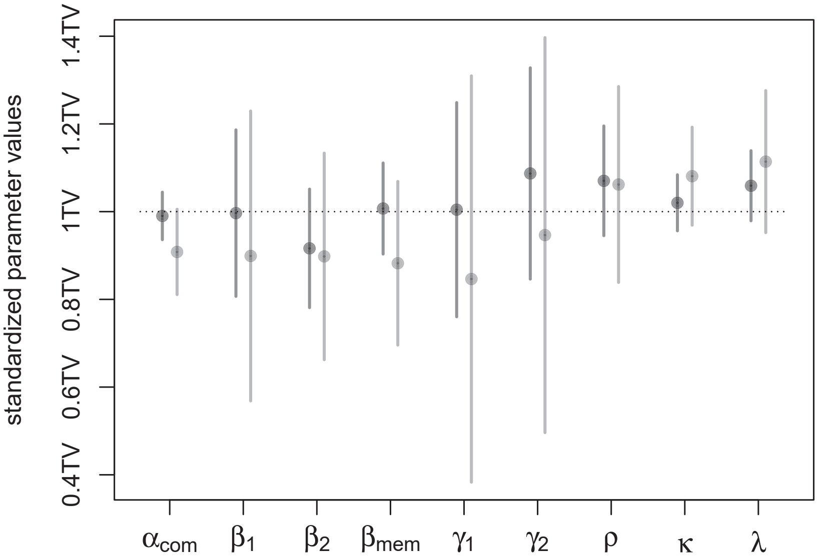


Fig. 5. Parameter estimates and their 95% confidence intervals for the combination trajectory on landscape 2, fitted with the combination model. Results for the complete trajectory (dark grey) compared to results for the same trajectory with 10% missing locations (light grey). Parameter estimates and confidence intervals are scaled by the true parameter values (TV): $\kappa = 5.5$, $\lambda = 1.6$, $\rho = 1$, $\alpha_{\text{res}} = -1$, $\alpha_{\text{mem}} = -4$, $\alpha_{\text{com}} = -5$, $\beta_1 = 1$, $\beta_2 = 2$, $\beta_{\text{mem}} = 0.03$, $\gamma_1 = 0.01$, γ_2=0.05.

B.3 Convergence and estimability issues

B.3.1 Estimability in cases of non-convergence

In each data set, about 18-20% of model fits contained one or more parameters, for which $\hat{R}$ was larger than 1.1, indicating non-convergence or non-mixing of the parallel chains. In many cases, we continued to run these chains for the double or triple amount of iterations, without ever seeing a major change in the chains' behaviour. Of course, we cannot exclude the possibility that after many more iterations (tens of thousands) the Markov chains would have finally reached convergence, or in case of non-mixing parallel chains would have switched their behaviour. However, the model fits, especially for the combination model, were both time-consuming (MCMC runs with two parallel chains could take 1-10 days, depending on model) and memory-intense (using approximately 1-5 GB RAM, depending on model). Considering that processing the three presented data sets required in total 240 model fits, we tried to reduce MCMC iterations to a reasonable amount, which in most cases lead to convergent and well-behaved Markov chains.

To understand convergence problems, we calculated data cloning estimability diagnostics and likelihood slices for selected trajectories. To obtain estimability diagnostics for a trajectory, we had to run the data cloning algorithm several times for increasing number of clones. This was even more computationally demanding than running data cloning for a single fixed number of clones. Therefore, we did not calculate estimability diagnostics for all model fits.

We performed estimability analysis on selected trajectories across all three data sets to understand a variety of phenomena.

In data set 2, large $\hat{R}$ values occurred in a matching model fit (fourth combination trajectory; see previous section). We inspected MCMC traces and posterior distributions. The two parallel chains in the MCMC did not mix, but each chain appeared to converge on its own (Fig. 7). This resulted in bimodal posterior distributions of the parameters. We calculated estimates from each of the chains separately and calculated their likelihood values. The estimates from one chain (red chain in Fig. 7), say $\hat{\boldsymbol{\theta}}_1$, were close to the true underlying values of the trajectory with $\log L(\hat{\boldsymbol{\theta}}_1) = -6783.94$. In comparison, true parameter values had slightly lower log-likelihood $\log L(\boldsymbol{\theta}_{\text{true}}) = -6788.582$. The estimates from the other chain (black chain in Fig. 7), say $\hat{\boldsymbol{\theta}}_2$, had a lower log-likelihood value of $\log L(\hat{\boldsymbol{\theta}}_2) = -7121.017$. It appears that the likelihood function has a local maximum at $\hat{\boldsymbol{\theta}}_2$. While the first chain found the higher peak, the second chain found the second, lower, peak and failed to move away from it. Because $L(\hat{\boldsymbol{\theta}}_2)$ was distinctly lower than $L(\hat{\boldsymbol{\theta}}_1)$, it did not appear that the chains' behaviour was due to an estimability problem. To confirm this, we calculated estimability diagnostics. This required a new model fit with varying number of clones. In this fit, all Markov chains converged and mixed well. All posterior variances decreased with increasing number of clones and `lambda.max` converged to zero with rate $\frac{1}{K}$, where K is the number of clones (Fig. 6).

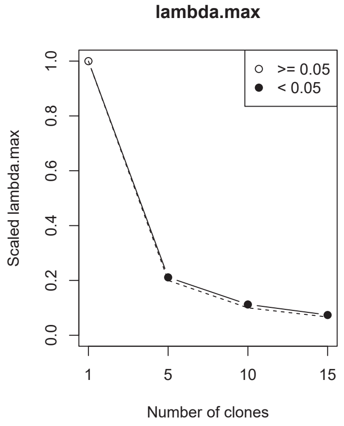


Fig. 6. Comparison of `lambda.max` (points and solid line) and the line $\frac{1}{k}$ (dotted) for a matching combination model fit that had mixing problems at first but converged in a second MCMC run. No estimability problems found.

We looked into estimability for three more model fits that did not converge the first time.

We analyzed a non-convergent model fit in data set 1, in which the combination model was fitted to a memory trajectory. When we calculated estimability diagnostics, all Markov chains converged and `lambda.max` behaved well and did not indicate any estimability problems.

We further analyzed estimability for a non-convergent fit in data set 1, where a memory model was fitted to a resource trajectory. Here, variances of parameter estimates decreased properly for the kernel parameters, however not for selection parameters, α_{mem} and β_{mem}, indicating estimability issues (Fig. 8). This means that the selection parameters of the memory model with respect to time since last visit could not be determined for the resource trajectory, which indeed did not truly contain an effect of this dynamic variable.

Because most convergence problems occurred when a more complex model was fitted to a null trajectory, we also examined a non-convergent fit of the combination model to a null trajectory in data set 2. Inspection of the non-convergent Markov chains showed that for most parameters, parallel chains did not mix but sampled different regions of the parameter space, resulting in bimodal posterior distributions. We separately calculated estimates and their likelihood values for the two chains, and the likelihood difference was smaller than 1. Posterior variances indicated estimability problems for all selection parameters, which are the parameters that were not relevant to the null trajectory (Fig. 9).

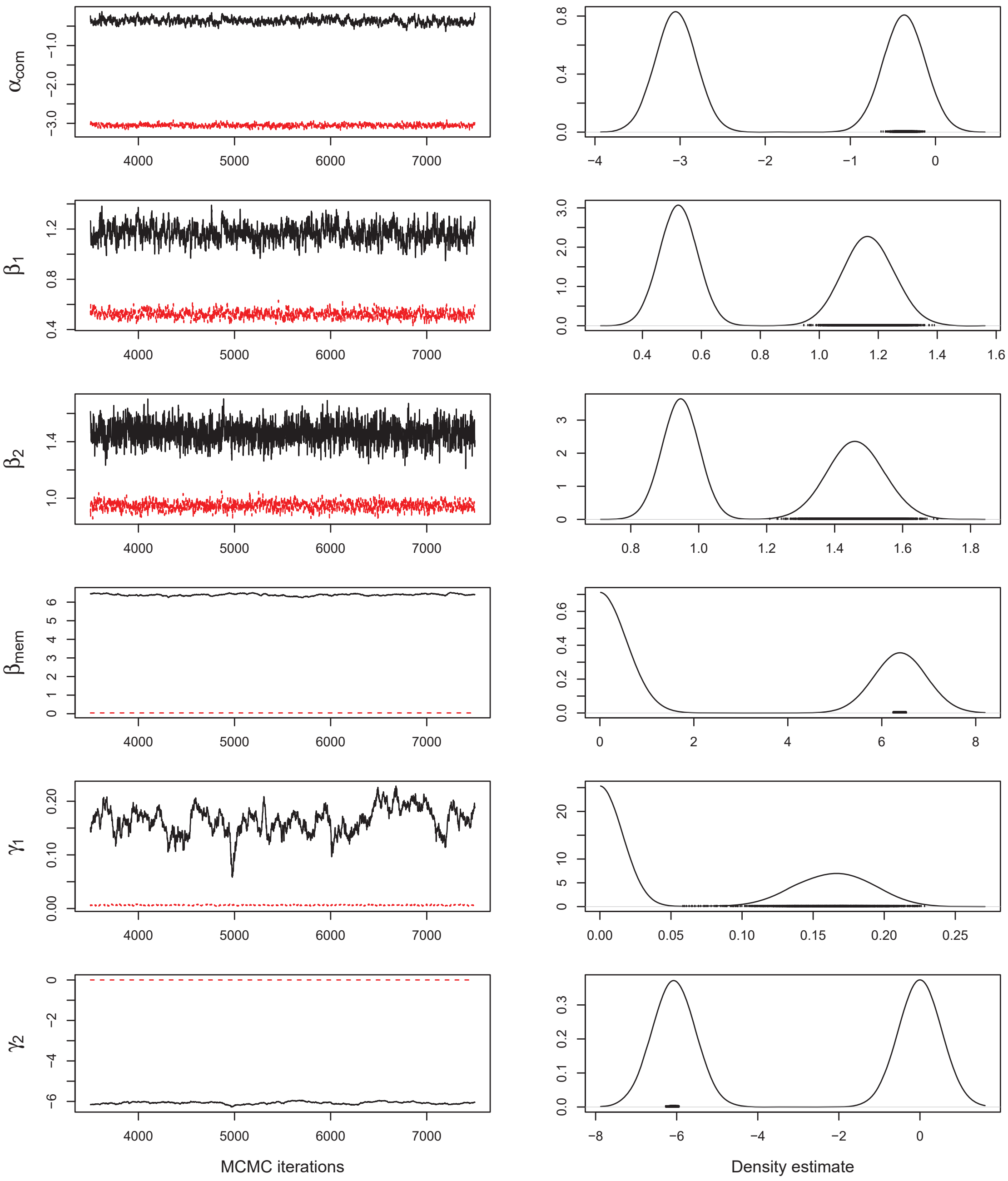


Fig. 7. MCMC results for the combination trajectory that had mixing problems when fitted with the matching combination model. Trace plots of MCMC iterations and density plots are shown for all parameters of the weighting function. The two parallel chains do not mix, but each appear to converge on their own. Estimates derived only from the red chain have higher likelihood value and than estimates derived from only the black chain. Estimates from the red chain are close to the true underlying values of the trajectory.

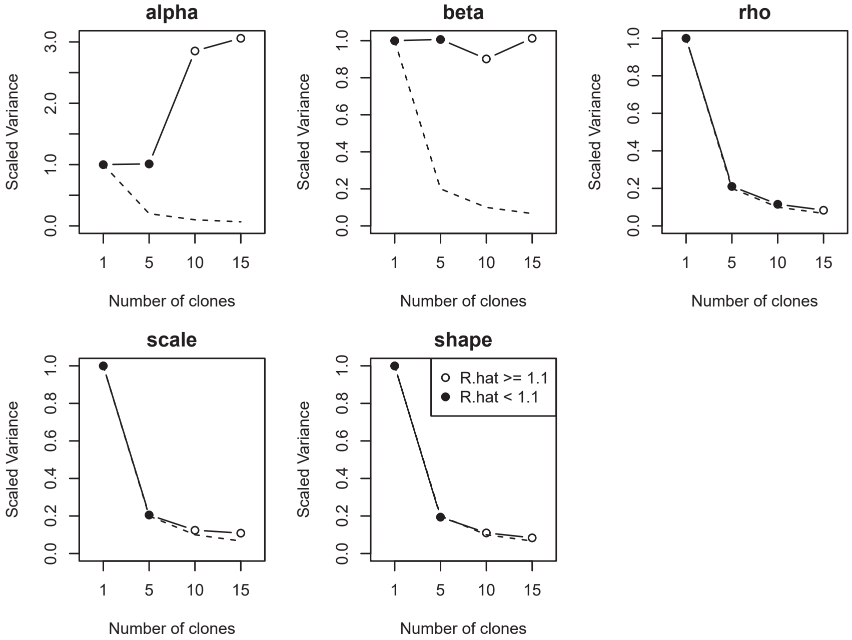


Fig. 8. Estimability diagnostics for a memory model fitted to a resource trajectory that did not converge in data set 1. The plots show variances of the posterior distributions for increasing number of clones. Non-decreasing variances of α_{res} and β_{mem} indicate estimability issues.

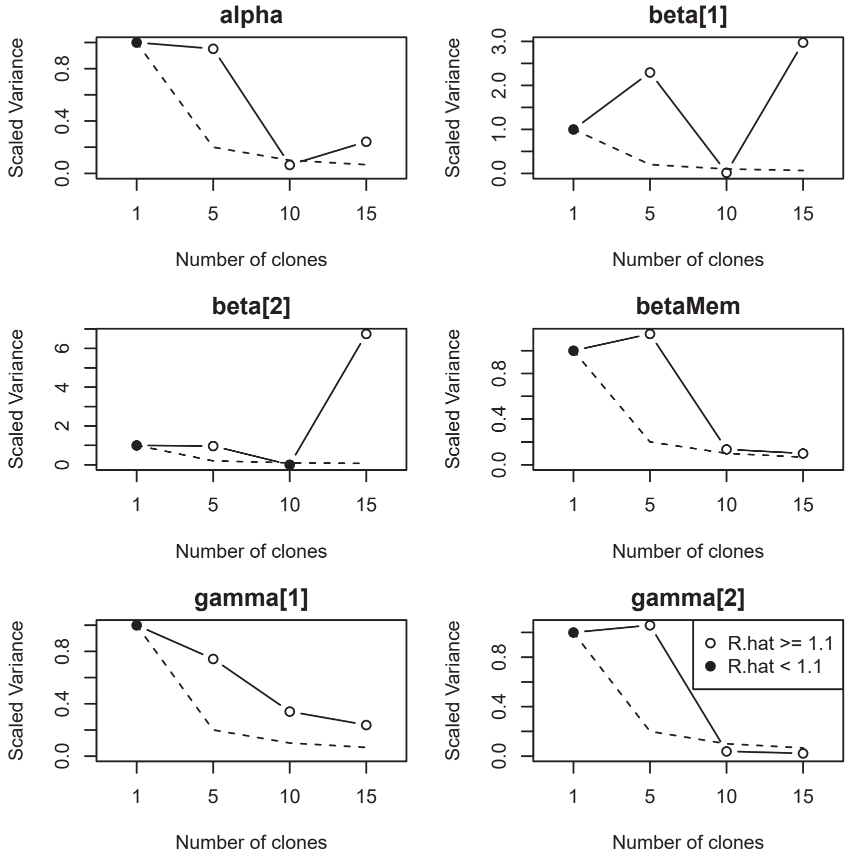


Fig. 9. Estimability diagnostics for a combination model fitted to a null trajectory that did not converge in data set 2. The plots show variances of the posterior distributions of selection parameters for increasing number of clones, which all indicate estimability problems.

B.3.2 Estimability in cases of large confidence intervals

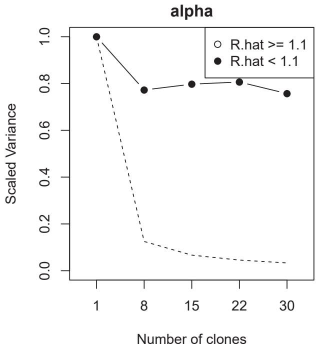


Fig. 10. Posterior variance of α_{res} for a matching resource model fit in data set 2, which shows estimability problems. The dotted line indicates the ideal line $\frac{1}{k}$.

In our analysis of parameter estimates and their confidence intervals in matching model fits, we found that even though MCMC runs converged, parameters of the weighting function occasionally had very large confidence intervals. One example is the matching fit of the fourth resource trajectory in data set 2, in which the estimate of α_{res} had an unusually high value, together with a large confidence interval. Estimability diagnostics showed decreasing posterior variances for all parameters except α_{res} (Fig. 10). We suspected that these results were caused by a ridge in the likelihood along α_{res}. We therefore calculated two-dimensional likelihood slices to confirm this. Fig. 11 shows two example slices, in which α_{res} and β_2 vary, whereas all remaining parameters are fixed. First, we fixed remaining parameters at their true values. The resulting surface over α_{res} and β_2 has a local maximum with log-likelihood value -7099.697 (panels (a-i) and (a-ii)). However, when we fixed the other parameters at their MLE values obtained from the model fit, the surface shows a ridge (panels (b-i) and (b-ii)). This ridge has a log-likelihood value of -7098.876, which is slightly higher than the local maximum of the other slice. The MCMC explores this area and moves along the ridge. It appears that the ridge has a very subtle maximum between -10 and -9, but it is so subtle that the MCMC extensively moves along the entire ridge.

In our model selection analysis, there was one matching model fit that converged but did not result in lowest BIC. This was a resource trajectory, for which the null model had lower BIC. In the matching fit with the resource model, estimates of selection parameters had high absolute values and large confidence intervals. When we calculated estimability diagnostics for a series of clones $K = 1, 5, 10, 15$, `lambda.max` showed signs of estimability problems (Fig. 12(a)). Estimates were very similar to the first MCMC run ($\hat{\alpha}_{\text{res}} = 10.4$, $\hat{\beta}_1 = -24$, $\beta_2 = -3.6$). However, because `lambda.max` generally decreased, we considered the possibility that we had not used enough clones. We therefore calculated estimability diagnostics for $K = 1, 8, 15, 22, 30$. In this run, α_{res} and β_1 showed good behaviour, however β_2 had high value and large confidence interval ($\hat{\alpha}_{\text{res}} = 1.3$, $\hat{\beta}_1 = -1.3$, $\beta_2 = 8.9$). Estimability diagnostics showed potential issues with β_2 (Fig. 12(b),(d)). To understand this further, we also looked at the likelihood function. We calculated log-likelihood values for the true underlying parameter values ($L_{\text{true}} = -6920$), the estimates from the first estimability run for $K = 15$ ($L_{1,k15} = -6915$), and the estimates from the second estimability run for $K = 30$ ($L_{2,k30} = -6918$). We compared these with the log-likelihood value for the model fit in which we fitted the null model to the trajectory ($L_{\text{null}} = -6921$). It appears that estimates from the first run approximate the MLE, whereas estimates from the last run with 30 clones arise from a local maximum with only slightly lower likelihood. When we plotted a likelihood slice for this model fit (fixing all parameters but β_2 at their estimates), we found a potential ridge in the likelihood for positive large values of β_2 (Fig. 12(c)). On the other hand, in the region of the estimates from the first run, we did not see any signs of ridges, however log-likelihood

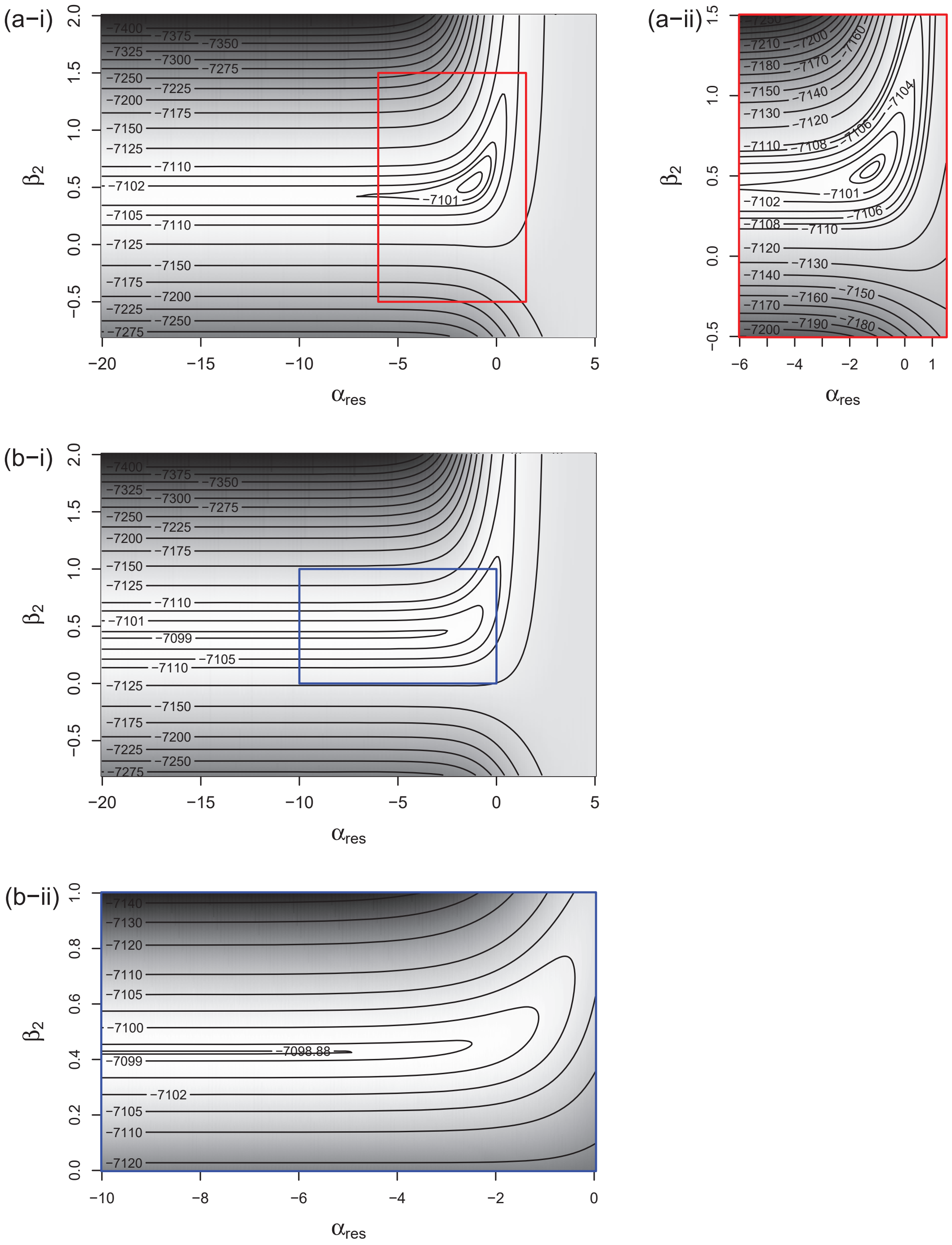


Fig. 11. Slices of the log-likelihood function for resource trajectory with estimability issues. Panels (a-i) and (a-ii) show the log-likelihood surface when all parameters are fixed at their true values except α_{res} and β_2. The surface shows a local peak. Panels (b-i) and (vii) show the log-likelihood surface when parameters instead are fixed at their MLE values. This surface shows a ridge and the log-likelihood value of this ridge is slightly higher than the local peak in panels (a-i) and (a-ii). Therefore, α_{res} cannot be uniquely estimated.

values did not vary much. From these tests, we concluded that the likelihood surface for this resource trajectory has a difficult structure for optimization, but a maximum exists in the region of selection parameters $\hat{\alpha}_{\text{res}} = 10.4$, $\hat{\beta}_1 = -24$, $\beta_2 = -3.6$. These parameter values result in a weighting function that is almost constant (Fig. 13(b)) and therefore model selection via BIC prefers the more parsimonious null model (Fig. 13(d)). Estimates from our second estimability run for $K = 30$ are closest to the true values (Fig. 13(a) and (c)), however, the likelihood is lower for these values.

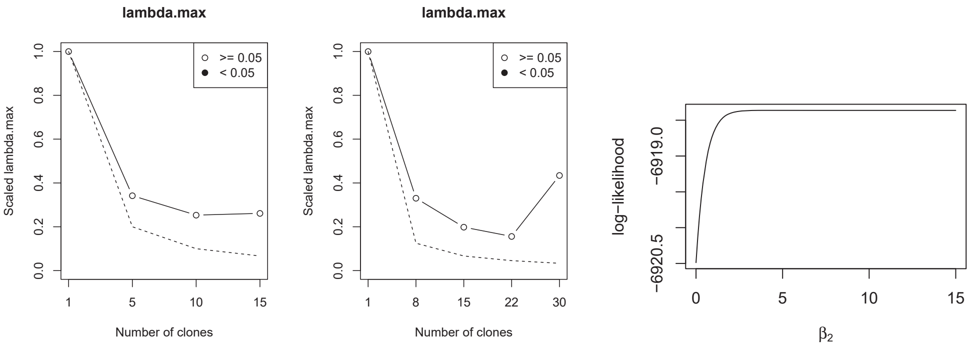
B.4 Conclusions about model fitting

In general, the results for the supplemental data are similar to the results for the data set presented in the main text of the paper. Although we designed the supplemental data set to include potentially more difficult estimation scenarios, our framework was able to detect effects of both resources and the dynamic variable time since last visit.

About 20% of data cloning MCMC runs did not converge the first time, and we did not achieve improvement by increasing the number of MCMC iterations. However, when we re-started certain model fits for estimability analysis, occasionally MCMC runs converged in this second run. From this experience, we recommend to rather re-start MCMC sampling completely instead of running more iterations, especially when traces show that parallel chains sample distinct regions of the parameter space, leading to bimodal posteriors. We recommend to additionally calculate estimability diagnostics for these model fits. If these indicate estimability problems for certain parameters, this may be an indicator that a model contains covariates that in fact did not influence the movement process. In this case, we recommend fitting alternative models or sub-models and comparing them via model selection.

We can relate convergence problems to two different phenomena of the likelihood function. First, the likelihood function had local maxima or ridges but still a unique global maximum. In such cases, single chains could occasionally fail to find the global maximum. This could be a potential difficulty with data cloning. In data cloning, every peak in the likelihood function is enhanced, including local maxima. If a chain by chance, e.g. via a ridge-like structure, reaches a local maximum, it may have difficulty moving away from it. This may also depend on the MCMC algorithm used. Other methods such as standard maximization of the likelihood function are not safe from this problem of local maxima either. It is thus for any method important to use multiple starting points or parallel chains. As second reason for non-convergent chains we found likelihood functions that had ridges or distinct multiple maxima, i.e. global maxima with almost the same likelihood value. These were clear cases of estimability problems, and were detected by data cloning estimability diagnostics. Any other method will fail in these cases too, either through non-convergence or results that indicate multiple possibilities for estimates (multiple maxima are found, bimodality of posteriors). If no problems are detected in these cases, this is even worse, because wrong conclusions are made.

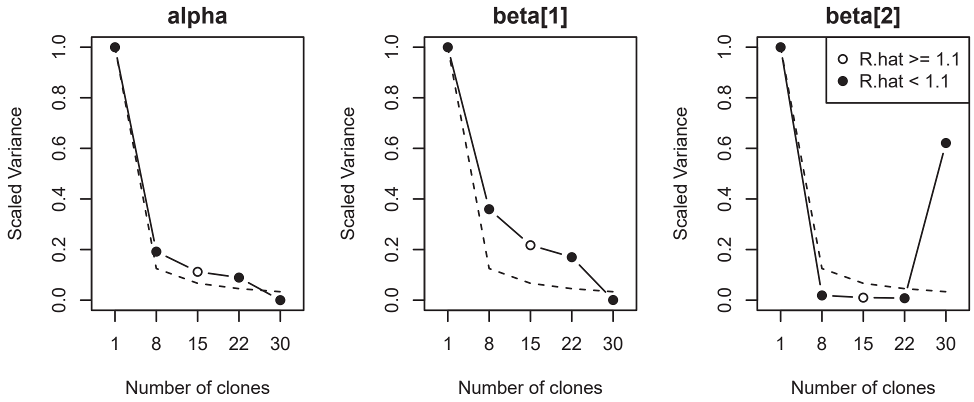
Most convergence problems occurred when a more complex model was fitted to a null trajectory. We suspect that this may be partly due to the form of the weighting function. In the logistic function, $\big(1 + \exp(-\alpha - \beta x)\big)^{-1}$, large values of the intercept α can cause the exponential function to almost vanish, leading to a nearly constant logistic function. Large selection parameters β can also have this effect. Therefore, if we fit a model that includes



(a) Largest posterior variance for the first estimability test.

(b) Largest posterior variance for the second estimability test with higher number of clones.

(c) Likelihood slice, all other parameters fixed at estimates obtained from second estimability test for $k = 30$.



(d) Separate posterior variances for selection parameters for the second estimability test with higher number of clones.

Fig. 12. Estimability diagnostics for the fourth matching resource model fit in data set 3. Two different series of MCMC runs with varying number of clones suggested difficulties with estimability. Region-wise calculation of the likelihood function confirmed a complex likelihood surface with a potential ridge (panel (c)) but nonetheless a slightly higher maximum far away from the true parameter values.

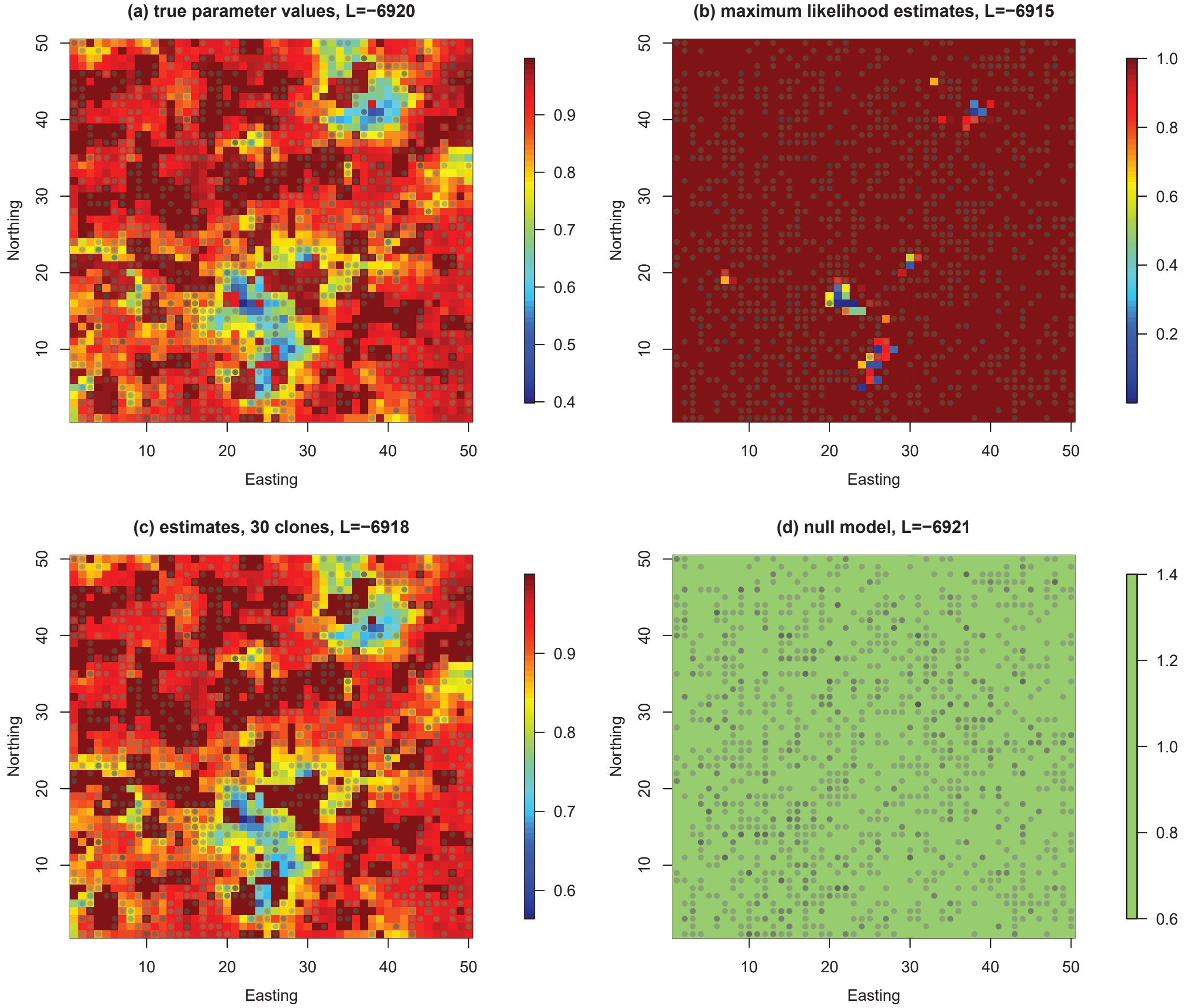


Fig. 13. Weighting function for different parameter estimates obtained for the resource trajectory on landscape four. For the matching fit with the resource model, we obtained different estimates from different MCMC runs due to a complex likelihood surface with local maxima. Panel (a): 'True' parameter values used in trajectory simulation. Panel (b): Estimates from a model fit with 15 clones. Panel (c): Estimates from an alternative model fit with 30 clones. Panel (d): Constant weighting function (null model). Gray dots are the locations of the trajectory. Darker dots correspond to multiple visits to a location. Their distribution across the entire home range also indicate a rather uniform use of space in accordance with weighting functions (b) and (d).

any kind of selection to a null trajectory, we can expect the likelihood function to have multiple maxima, ridges or plateaus, especially in those regions of the parameter space where parameters of the weighting function are large. Therefore, we are not alarmed that many of these model fits did not converge in our analysis. Via estimability diagnostics, we have the ability to detect such situations. As soon as trajectories contained at least one effect (either resources or time since last visit), convergence problems occurred less frequently. However, this phenomenon of the logistic function may give reason to also consider alternative forms of the weighting function that do not experience this problem.

In matching model fits, we occasionally observed unusually large deviations of estimates from true parameter values or large confidence intervals. From our experience, we suspect that this is mainly due to estimability issues (e.g. ridges in the likelihood). These could occur due to the stochasticity of our data simulation. Each trajectory is a realization of a stochastic process. In most cases, we expect trajectories to realize a behaviour according to the parameter values used for the simulations. Still, we expect to see also cases that are less well behaved.

References

Brooks, S.P. & Gelman, A. (1998) General methods for monitoring convergence of iterative simulations. *Journal of Computational and Graphical Statistics*, **7**, 434–455.

Casella, G. & Berger, R. (2001) *Statistical Inference*. Cengage Learning, 2 edition.

Haran, M. (2011) Gaussian random field models for spatial data. *Markov chain Monte Carlo Handbook Eds Brooks, SP, Gelman, AE Jones, GL and Meng, XL*, pp. 449–478. Chapman and Hall/CRC.

Hiebeler, D. (2000) Populations on fragmented landscapes with spatially structured heterogeneities: landscape generation and local dispersal. *Ecology*, **81**, 1629–1641.

Lele, S.R., Nadeem, K. & Schmuland, B. (2010) Estimability and likelihood inference for generalized linear mixed models using data cloning. *Journal of the American Statistical Association*, **105**, 1617–1625.

Ponciano, J.M., Taper, M.L., Dennis, B. & Lele, S.R. (2009) Hierarchical models in ecology: confidence intervals, hypothesis testing, and model selection using data cloning. *Ecology*, **90**, 356–362.

Robert, C.P. (1993) Prior feedback: Bayesian tools for maximum likelihood estimation. *Computational Statistics*, **8**, 279–294.

Schlather, M., Menck, P., Singleton, R., Pfaff, B. & R Core team (2013) *RandomFields: Simulation and Analysis of Random Fields*.

Solymos, P. (2010) dclone: Data Cloning in R. *The R Journal*, **2**, 29–37.