Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 227 (2007) 353–375
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

Generator estimation of Markov jump processes ☆

P. Metzner *, E. Dittmer, T. Jahnke, Ch. Schütte

Institute of Mathematics II, Free University of Berlin, Arnimallee 6, D-14195 Berlin, Germany

Received 8 March 2007; received in revised form 26 July 2007; accepted 27 July 2007
Available online 14 August 2007

Abstract

Estimating the generator of a continuous-time Markov jump process based on incomplete data is a problem which arises in various applications ranging from machine learning to molecular dynamics. Several methods have been devised for this purpose: a quadratic programming approach (cf. [D.T. Crommelin, E. Vanden-Eijnden, Fitting timeseries by continuous-time Markov chains: a quadratic programming approach, J. Comp. Phys. 217 (2006) 782–805]), a resolvent method (cf. [T. Müller, Modellierung von Proteinevolution, PhD thesis, Heidelberg, 2001]), and various implementations of an expectation-maximization algorithm ([S. Asmussen, O. Nerman, M. Olsson, Fitting phase-type distributions via the EM algorithm, Scand. J. Stat. 23 (1996) 419–441; I. Holmes, G.M. Rubin, An expectation maximization algorithm for training hidden substitution models, J. Mol. Biol. 317 (2002) 753–764; U. Nodelman, C.R. Shelton, D. Koller, Expectation maximization and complex duration distributions for continuous time Bayesian networks, in: Proceedings of the twenty-first conference on uncertainty in AI (UAI), 2005, pp. 421–430; M. Bladt, M. Sørensen, Statistical inference for discretely observed Markov jump processes, J.R. Statist. Soc. B 67 (2005) 395–410]). Some of these methods, however, seem to be known only in a particular research community, and have later been reinvented in a different context. The purpose of this paper is to compile a catalogue of existing approaches, to compare the strengths and weaknesses, and to test their performance in a series of numerical examples. These examples include carefully chosen model problems and an application to a time series from molecular dynamics.

MSC: 60C40; 60J22; 60J75; 60J35; 60M05

PACS: 02.30.Zz; 02.50.Ga; 05.45.Tp; 36.20.Ey

Keywords: Markov jump process; Generator estimation; Embedding problem; EM-algorithm; Maximum likelihood

☆ Supported by the DFG Research Center MATHEON "Mathematics for Key Technologies" (FZT86) in Berlin.
* Corresponding author.
E-mail addresses: metzner@math.fu-berlin.de (P. Metzner), dittmer@math.fu-berlin.de (E. Dittmer), jahnke@math.fu-berlin.de (T. Jahnke), schuette@math.fu-berlin.de (Ch. Schütte).

0021-9991/$ - see front matter © 2007 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2007.07.032

1. Introduction

Let $\{X(t), t \geqslant 0\}$ be a continuous-time Markov jump process on a finite state space $S \cong \{1, \ldots, d\}$. Only time-homogeneous Markov processes will be considered, i.e. we assume that

$$\mathcal{P}(X(t+\tau) = j | X(t) = i) = \mathcal{P}(X(\tau) = j | X(0) = i)$$

for all states $i,j \in S$ and all $t,\tau \geqslant 0$. The transition matrix of $\{X(t), t \geqslant 0\}$ is the time-dependent matrix

$$P(t) = (p_{ij}(t))_{i,j} \in \mathbb{R}^{d\times d}, \quad p_{ij}(t) = \mathcal{P}(X(t) = j | X(0) = i)$$

containing the transition probabilities $p_{ij}(t)$. If the limit

$$L = \lim_{t\to 0} \frac{P(t) - Id}{t}$$

exists, then the transition matrix can be expressed as the matrix exponential

$$P(t) = \exp(tL) = \sum_{k=0}^{\infty} \frac{t^k}{k!} L^k$$

and L is called the *infinitesimal generator* of the Markov process $\{X(t), t \geqslant 0\}$. A matrix $L \in \mathbb{R}^{d\times d}$ generates a continuous-time Markov process if and only if all off-diagonal entries are nonnegative and the sum over each row equals zero. The set of generators will be denoted by

$$\mathfrak{G} = \left\{ L = (l_{ij})_{i,j} \in \mathbb{R}^{d\times d} : l_{ij} \geqslant 0 \quad \text{for all } i \neq j, \quad l_{ii} = -\sum_{j\neq i} l_{ij} \right\}. \tag{1}$$

In this article, we consider the problem of how to determine the generator if only a finite sampling $Y = \{y_0 = X(t_0), \ldots, y_N = X(t_N)\}$ of a process at discrete times $t_0, t_1, \ldots, t_N$ is available. Several difficulties must be taken into account. First, from a finite number of samples it is impossible to tell if the underlying process is actually Markovian. Second, it is not clear if the observed data originates indeed from discrete samples of a continuous-time Markov chain with some generator L, or rather from a discrete-time Markov chain which cannot be embedded into a time-continuous counterpart. In the latter case, a generator does not exist because the transition matrix of the discrete chain does not belong to the set

$$\mathfrak{P} = \{P \in \mathbb{R}^{d\times d} : \text{there is a } L \in \mathfrak{G} \text{ such that } P = \exp(L)\}.$$

It is well-known that $\mathfrak{P}$ is a subset of all stochastic matrices, but the so-called *embedding problem*, i.e. the question what characterizes the elements of $\mathfrak{P}$, is widely open for $d > 3$ (cf. [1,2] and references therein). A third difficulty is the fact that the matrix exponential function *is not injective* if the eigenvalues of the generator are complex. Hence, some matrices $P \in \mathfrak{P}$ can be represented as $P = \exp(L) = \exp(\overline{L})$ with two different generators $L \neq \overline{L}$. And finally, the question whether the time points t_n of the observations are equidistant plays an important role. In case of a constant time lag $\tau = t_{n+1} - t_n$ an estimate of the transition matrix $P(\tau)$ is available by counting the number of transitions between each pair of states, but in case of variable time lags the sampled data is typically not sufficient for reasonable approximations of the transition matrix.

Due to these problems the above question has to be modified: how can we find the generator that "agrees best" with a finite observation $Y = \{y_0 = X(t_0), \ldots, y_N = X(t_N)\}$ of a process? Several methods have been proposed for this purpose [2,19,18,3,1], but since the problem has been investigated in the context of rather different applications, some of these papers apparently remained unknown outside the corresponding research community or were at least not cited. Some methods have later been reinvented, such as the expectation-maximization algorithm introduced in [19] in 1996 and later by [1] in 2005. This may partly be due to the fact that even the terminology varies largely form paper to paper – for example, the generator is often called (transition) rate matrix (cf. [18]), substitution matrix (cf. [20]), intensity matrix (cf. [1]), or "matrix of transition intensities" (cf. [17]).

In this article, we present a concise – but not necessarily complete – review of approaches to generator estimation. We explain how each method works, discuss their pros and cons, evaluate their performance by numerical examples, and explain in which sense the "agreement" of the approximative generator with the data has to be understood in each case. In Section 2, we briefly sketch the resolvent method advocated, e.g., in [3].

However, since the output of this method is in general not a generator matrix, it will later be excluded from the comparison. Section 3 resumes the quadratic programming approach introduced in [2]. In Section 4, we discuss several methods for the computation of maximum likelihood estimators and the expectation-maximization algorithms proposed, e.g., in [19,17,18,1]. Furthermore, we derive a version of the EM-algorithm which is adapted to the reversible case. Numerical examples are presented in Section 5. The performance of the methods is illustrated by carefully chosen model problems and by an application to a large system arising from molecular dynamics. In the last section, we discuss and summarize the numerical results and give an outlook on a generalization of an enhanced expectation-maximization method to the case of varying time lags.

2. Resolvent method

For any generator $L \in \mathfrak{G}$ the parameter-dependent matrix

$$R(\alpha) = (\alpha I - L)^{-1}, \quad \alpha > 0 \tag{2}$$

is called the resolvent of L. The inverse exists for all $\alpha > 0$ since the real parts of the eigenvalues of a generator $L \in \mathfrak{G}$ are non-positive. An alternative formula representing the resolvent in terms of the transition matrix $P(t) = \exp(tL)$ is given by the Laplace transform

$$R(\alpha) = \int_0^\infty \exp(-\alpha t) P(t) \mathrm{d}t, \quad \alpha > 0. \tag{3}$$

The existence of the integral is due to the fact that $\|P(t)\| = 1$ for all $t \geqslant 0$, and the equivalence of (3) and (2) follows from

$$(\alpha I - L) R(\alpha) = -\int_0^\infty \frac{\mathrm{d}}{\mathrm{d}t} (\exp(-\alpha t) P(t)) \mathrm{d}t = I.$$

The main idea of the resolvent method is to approximate the resolvent using its integral representation (3) and then to estimate the underlying generator via the identity

$$L = \alpha Id - R^{-1}(\alpha). \tag{4}$$

Computing the integral in (3), however, requires an approximation of the transition matrix $P(t)$. Suppose that the process $X(t)$ has been observed at equidistant time points $t_n = n\tau$ with some fixed time lag $\tau > 0$ and $n = 0, \ldots, N$. Let

$$c_{ij}^{(k)} = \sum_{n=0}^{N-k} \chi(X(t_n) = i) \chi(X(t_{n+k}) = j) \tag{5}$$

be the number of observed transitions from state i to state j within the time interval of length $k\tau$. Here and below, χ denotes the characteristic function. The matrix $C^{(k)} = (c_{ij}^{(k)})_{i,j} \in \mathbb{N}^{d \times d}$ is called the frequency matrix with respect to the time interval $[0, k\tau]$, and a simple estimate $\widetilde{P}^{(k)} \approx P(t_k)$ is provided by

$$\widetilde{P}^{(k)} = \left(\tilde{p}_{ij}^{(k)}\right)_{i,j} \quad \text{with entries } \tilde{p}_{ij}^{(k)} = \frac{c_{ij}^{(k)}}{\sum_{j=1}^d c_{ij}^{(k)}}. \tag{6}$$

In the approach of [5,3] these estimates are used to approximate $P(t)$ in the interval $[t_n, t_{n+1}]$ by linear interpolation:

$$P(t) \approx P(t_n) + (t - t_n) \frac{P(t_{n+1}) - P(t_n)}{\tau} \approx \widetilde{P}^{(n)} + (t - t_n) \frac{\widetilde{P}^{(n+1)} - \widetilde{P}^{(n)}}{\tau}.$$

Substituting this into the integral representation (3) gives

$$\begin{aligned} R(\alpha) &= \sum_{n=0}^{m-1} \int_{t_n}^{t_{n+1}} \exp(-\alpha s) P(s) \, \mathrm{d}s + \int_{t_m}^\infty \exp(-\alpha s) P(s) \, \mathrm{d}s \\ &\approx \sum_{n=0}^{m-1} \int_{t_n}^{t_{n+1}} \exp(-\alpha s) \left(\widetilde{P}^{(n)} + (s - t_n) \frac{\widetilde{P}^{(n+1)} - \widetilde{P}^{(n)}}{\tau} \right) \mathrm{d}s + \int_{t_m}^\infty \exp(-\alpha s) P(t_m) \, \mathrm{d}s. \end{aligned} \tag{7}$$

Since all integrals in (7) can be solved analytically, this yields an approximation $\widetilde{R}(\alpha) \approx R(\alpha)$ to the resolvent. If $\widetilde{R}(\alpha)$ is invertible, then Eq. (4) yields an estimate $\widetilde{L}^{(\alpha)} = \alpha Id - \widetilde{R}^{-1}(\alpha)$ for the generator. Of course, the estimate depends on the particular choice of α, but the optimal value of α can be determined by a maximum likelihood approach; see [5,3] for details.

It can easily be shown that for any α the entries $\tilde{l}_{ii}^{(\alpha)}$ of the estimated generator $\widetilde{L}^{(\alpha)}$ satisfy the condition $\tilde{l}_{ii}^{(\alpha)} = -\sum_{j\neq i} \tilde{l}_{ij}^{(\alpha)}$. However, $\widetilde{L}^{(\alpha)}$ is in general *not* a generator in the sense of (1), because $\widetilde{L}^{(\alpha)}$ can contain negative or even complex off-diagonal elements. This happens if some of the estimated transition matrices $\widetilde{P}^{(n)}$ do not belong to the set $\mathfrak{P}$. In [3], this obstacle did not appear because the resolvent method was applied to problems where the transition matrices could assumed to be *calibrated* (i.e. close to identity in some sense). In a general situation, however, the fact that $\widetilde{L}^{(\alpha)} \notin \mathfrak{G}$ is a severe drawback of the resolvent method.

3. Quadratic optimization method

In contrast to the resolvent method, the approach introduced by Crommelin and Vanden-Eijnden [2] yields an estimate that *does* belong to the set $\mathfrak{G}$. As in the previous chapter, first an approximative transition matrix $\widetilde{P}^{(1)} \approx P(t_1) = P(\tau)$ is computed by Eq. (6). Now suppose an eigendecomposition

$$\widetilde{P}^{(1)} = U \Lambda U^{-1} \tag{8}$$

with a diagonal matrix $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_d)$ containing the eigenvalues exists, and that $\lambda_k \neq 0$ for all k. (Note that U^{-1} can be obtained without explicit matrix inversion since its rows are the left eigenvectors of $\widetilde{P}^{(1)}$.) Then, the matrix

$$\widetilde{L} = UZU^{-1} \quad \text{with } Z = \mathrm{diag}(z_1, \ldots, z_d), \quad z_k = \frac{\log(\lambda_k)}{\tau} \tag{9}$$

can be defined, and the approximative transition matrix can be expressed in terms of the matrix exponential

$$\exp(\tau \widetilde{L}) = \exp(U \log(\Lambda) U^{-1}) = U \Lambda U^{-1} = \widetilde{P}^{(1)}.$$

In spite of this relation, $\widetilde{L}$ cannot be considered as a reasonable estimate for the generator because $\widetilde{L} \notin \mathfrak{G}$ in many cases. In order to find an estimate with the correct structural properties, Crommelin and Vanden-Eijnden propose to compute the generator $\widetilde{L} \in \mathfrak{G}$ which agrees best with the eigendecomposition (9). This is motivated by the fact that many properties of a continuous-time Markov chain (such as, e.g., its stationary distribution) depend strongly on the eigenvalues and eigenvectors of its generator. Therefore, in [2] the generator is estimated by solving the quadratic minimization problem

$$\widetilde{L}_{\mathrm{QP}} = \arg\min_{L \in \mathfrak{G}} \sum_{k=1}^{d} (\alpha_k |U_k^{-1} L - z_k U_k^{-1}|^2 + \beta_k |L U_k - z_k U_k|^2 + \gamma_k |U_k^{-1} L U_k - z_k|^2) \tag{10}$$

where U_k denotes the kth column of U, U_k^{-1} is the kth row of U^{-1}, and

$$\alpha_k = a_k |z_k U_k^{-1}|^{-2}, \quad \beta_k = b_k |z_k U_k|^{-2} \quad \text{and} \quad \gamma_k = c_k |z_k|^{-2}$$

are weights with suitably chosen coefficients a_k, b_k, c_k. The problem (10) can be solved with a standard quadratic optimizer such as the Matlab `quadprog` command after reformulating (10) as

$$\widetilde{L}_{\mathrm{QP}} = \arg\min_{L \in \mathfrak{G}} \frac{1}{2} \langle L, HL \rangle + \langle F, L \rangle + E_0$$

with a tensor $H \in \mathbb{R}^{d\times d\times d\times d}$ and a matrix $F \in \mathbb{R}^{d\times d}$; see [2] for details. If d is so large that the tensor H cannot be stored, the problem (10) can still be solved with `quadprog`, but this requires a function for the evaluation of Hv for arbitrary v without composing H explicitly.

4. The maximum likelihood method

In this section, we explain in detail the maximum likelihood method introduced in [19] and elaborated further in [1]. The idea behind the maximum likelihood estimation (MLE) method is to find a generator $\widetilde{L}$ such that it maximizes the *discrete likelihood* of the given time series.

4.1. Continuous and discrete likelihood functions

The basis objects in the MLE-method is the continuous and discrete likelihood function. Suppose that the Markov jump process $X(t)$ has been observed continuously in a certain time interval $[0, T]$. Let the random variable $R_i(T)$ be the time the process spent in state i before time T

$$R_i(T) = \int_0^{\mathrm{T}} \chi(X(s) = i)\mathrm{d}s$$

and denote by $N_{ij}(T)$ the number of transitions from state i to state j in the time interval $[0, T]$. The *continuous time likelihood function* $\mathcal{L}_c$ of an observed trajectory $\{X_t\colon 0 \leqslant t \leqslant T\}$ is given by [1]

$$\mathcal{L}_c(L) = \prod_{i=1}^{d} \prod_{j\neq i} l_{ij}^{N_{ij}(T)} \exp(-l_{ij}R_i(T)), \quad L = (l_{ij}). \tag{11}$$

By definition, the maximum likelihood estimator (MLE) $\widetilde{L}$ maximizes the likelihood function (11). Exploiting the monotonicity of the log-function, $\widetilde{L}$ is also the maximizer of

$$\log \mathcal{L}_c(L) = \sum_{i=1}^{d} \sum_{j\neq i} [N_{ij}(T)\log(l_{ij}) - l_{ij}R_i(T)], \tag{12}$$

i.e. $\widetilde{L}$ is the null of the partial derivatives of $\log \mathcal{L}_c(L)$ with respect to l_{ij} and the Hessian matrix of $\log \mathcal{L}_c(L)$ evaluated at $\widetilde{L}$ is negative definite. A short calculation shows

$$\frac{\partial \log \mathcal{L}_c(\widetilde{L})}{\partial l_{ij}} = 0 \Longleftrightarrow \tilde{l}_{ij} = \frac{N_{ij}(T)}{R_i(T)} \tag{13}$$

and

$$\frac{\partial \log \mathcal{L}_c(\widetilde{L})}{\partial l_{ij}\partial l_{kl}} = -\frac{N_{ij}(T)}{\tilde{l}_{ij}^2}\chi((i,j) = (k,l)).$$

In the case where the process has only been observed at discrete time points $0 = t_0 < t_1 < \ldots < t_N = T$ the *discrete log-likelihood function* $\mathcal{L}_d$ of a time series $Y = \{y_0 = X(t_0), \ldots, y_N = X(t_N)\}$ is given in terms of the transition matrix $P(t) = \exp(tL)$

$$\mathcal{L}_d(L) = \prod_{k=0}^{n-1} [p_{y_k,y_{k+1}}(\tau_k)] \tag{14}$$

where $\tau_k = t_{k+1} - t_k$ is the time lag between two consecutive observations and $p_{y_k,y_{k+1}}(\tau_k)$ is the probability that the process makes a transition from state y_k to the state y_{k+1} in time τ_k. The discrete likelihood function (14) simplifies further under the assumption that the time lags $\tau_k = \tau$ are constant for $\tau > 0$,

$$\mathcal{L}_d(L) = \prod_{i,j=1}^{d} [p_{ij}(\tau)]^{c_{ij}} \tag{15}$$

where $c_{ij} = c_{ij}^{(1)}$ is the frequency of transitions from state i to state j in the discrete time Markov chain $Y = \{y_0, \ldots, y_N\}$ (compare (5)). Even for this simplified case, the derivative of (15) with respect to the entries of L

$$\frac{\partial}{\partial L} \log \mathcal{L}_d(L) = \sum_{n=1}^{\infty} \sum_{k=1}^{n} \frac{\tau^n}{n!} (L^{\mathrm{T}})^{k-1} Z (L^{\mathrm{T}})^{n-k} \quad \text{with } Z = (z_{ij})_{i,j\in S}, z_{ij} = c_{ij}/\exp(\tau L)_{ij}$$

has such a complicated form that the null cannot be found analytically. Hence no analytical expression for the MLE with respect to L is available.

4.2. Likelihood approach revisited

In the likelihood approach, a generator $\widetilde{L}$ for a given time series is determined such that $\widetilde{L}$ maximizes the discrete likelihood function (14) for the time series. As pointed out in the previous section the discrete likelihood function $\mathcal{L}_d$ does not permit an analytical maximum likelihood estimator. On the other hand, the MLE (13) for a continuous time observation can be obtained analytically but for an incomplete observation the information between two consecutive observations is *hidden* and, hence, the observables $R_i(T)$ and $N_{ij}(T)$ are unknown. In this situation the *expectation-maximization algorithm* (EM-algorithm) is a natural choice because it allows iteratively to approximate a local maximum of $\mathcal{L}_d$ by computing the expectation values of $R_i(T)$ and $N_{ij}(T)$ given the data and a generator guess. To be more precise, an iteration in the EM-algorithm consists of an expectation step (E-step) and a maximization step (M-step). In the E-step, the conditional expectations of the unknown parts in (11) with respect to the given data and a current guess $\widetilde{L}$ of the MLE are computed, i.e. $\mathbb{E}_{\widetilde{L}}[R_i(T)|Y]$ and $\mathbb{E}_{\widetilde{L}}[N_{ij}(T)|Y]$. In the maximization step, a new "guess" of a MLE is constructed via the maximizer (13) by replacing again the unobserved parts by their respective conditional expectations.

To formalize things, define the *conditional log-likelihood function*

$$\mathcal{G}(L;\widetilde{L}) = \mathbb{E}_{\widetilde{L}}[\log \mathcal{L}_c(L)|Y] \tag{16}$$

where $\mathbb{E}_{\widetilde{L}}$ denotes the conditional expectation with respect to a generator $\widetilde{L}$. Then, the EM-algorithm basically works as presented in Algorithm 1.

Algorithm 1 General EM-algorithm

Input: **Time series** $Y = \{y_0 = X(t_0), \ldots, y_N = X(t_N)\}$, initial guess of generator L_0.
Output: **MLE** $\widetilde{L}$.
(1) **Set** $\widetilde{L} := L_0$.
(2) **Expectation step (E-step):**
Compute the function $\mathcal{G}(L;\widetilde{L})$.
(3) **Maximization step (M-Step):**
$\widetilde{L} = \arg\max_L \mathcal{G}(L;\widetilde{L})$
(4) **Go to Step (2).**

Let $\widetilde{L}_0, \widetilde{L}_1, \widetilde{L}_2, \ldots$ be a sequence of generators obtained via the EM-algorithm. Dempster, Laird, and Rubin proved in [6] that an increase in $\mathcal{G}$ implies an increase in the discrete likelihood function

$$\mathcal{L}_d(\widetilde{L}_{k+1}) \geqslant \mathcal{L}_d(\widetilde{L}_k).$$

For our particular likelihood function (11) we obtain

$$\mathcal{G}(L;L_0) = \sum_{i=1}^{d}\sum_{j\neq i} \log(l_{ij})\mathbb{E}_{L_0}[N_{ij}(T)|Y] - \sum_{i=1}^{d}\sum_{j\neq i} l_{ij}\mathbb{E}_{L_0}[R_i(T)|Y] \tag{17}$$

and, consequently, the maximizer of (17) is given by

$$\tilde{l}_{ij} = \frac{\mathbb{E}_{L_0}[N_{ij}(T)|Y]}{\mathbb{E}_{L_0}[R_i(T)|Y]} \quad \text{for all } i \neq j. \tag{18}$$

The non-trivial task which remains is to evaluate the conditional expectations $\mathbb{E}_{L_0}[N_{ij}(T)|Y]$ and $\mathbb{E}_{L_0}[R_i(T)|Y]$, respectively. The first step towards their computation is the observation that by the Markov property, the homogeneity of the Markov jump process and a constant time lag τ the conditional expectations in (17) can be expressed as sums

$$\begin{aligned}
\mathbb{E}_{L_0}[R_i(T)|Y] &= \sum_{k=1}^{d}\sum_{l=1}^{d} c_{kl}\mathbb{E}_{L_0}[R_i(\tau)|X(\tau)=l, X(0)=k],\\
\mathbb{E}_{L_0}[N_{ij}(T)|Y] &= \sum_{k=1}^{d}\sum_{l=1}^{d} c_{kl}\mathbb{E}_{L_0}[N_{ij}(\tau)|X(\tau)=l, X(0)=k].
\end{aligned} \tag{19}$$

Next, the conditional expectations in the right hand sides in (19) can be decomposed further by using the identities

$$\begin{aligned}\mathbb{E}_L[R_i(t)|X(t)=l,X(0)=k] &= \frac{\mathbb{E}_L[R_i(t)\chi(X(t)=l)|X(0)=k]}{p_{kl}(t)},\\ \mathbb{E}_L[N_{ij}(t)|X(t)=l,X(0)=k] &= \frac{\mathbb{E}_L[N_{ij}(t)\chi(X(t)=l)|X(0)=k]}{p_{kl}(t)}.\end{aligned} \tag{20}$$

Finally, the authors in [19,1] realized that the auxiliary functions defined by

$$\begin{aligned}M^i_{kl}(t) &:= \mathbb{E}_L[R_i(t)\chi(X(t)=l)|X(0)=k],\\ F^{ij}_{kl}(t) &:= \mathbb{E}_L[N_{ij}(t)\chi(X(t)=l)|X(0)=k]\end{aligned} \tag{21}$$

satisfy systems of ordinary differential equations (ODEs). For example, let $i,j \in S$ be fixed. Then the vectors $M^i_k(t) = (M^i_{k1}(t),\ldots,M^i_{kd}(t))$ and $F^{ij}_k(t) = (F^{ij}_{k1}(t),\ldots,F^{ij}_{kd}(t))$ satisfy the two systems of ODEs

$$\begin{aligned}&\frac{\mathrm{d}}{\mathrm{d}t}M^i_k(t) = M^i_k(t)L + A^i_k(t), \quad M^i_k(0)=0 \quad \text{with } A^i_k(t) = p_{ki}(t)e_i,\\ &\frac{\mathrm{d}}{\mathrm{d}t}F^{ij}_k(t) = F^{ij}_k(t)L + A^{ij}_k(t), \quad F^{ij}_k(t)=0 \quad \text{with } A^{ij}_k(t) = l_{ij}p_{ki}(t)e_j,\end{aligned} \tag{22}$$

where e_i and e_j are the i-th and j-th unit vectors. To summarize, the computation of the function $\mathcal{G}(L;\widetilde{L})$ in the E-step reduces to solving the systems of ODEs given in (22). Solving these ODEs numerically, however, causes prohibitive computational costs when the number of states of the system is large. Another option is to approximate the matrix-exponentials which are involved in the analytic solutions of (22)

$$\begin{aligned}M^i_k(t) &= \int_0^t A^i_k(s)\exp((t-s)L)\mathrm{d}s,\\ F^{ij}_k(t) &= \int_0^t A^{ij}_k(s)\exp((t-s)L)\mathrm{d}s\end{aligned} \tag{23}$$

via the so-called uniformization method [7]. Choose $\alpha = \max_{i=1,\ldots,d}\{-l_{ii}\}$, and define $B = I + \alpha^{-1}L$. Then, e.g., $M^i(t) = (M^i_{kl}(t))_{k,l\in S}$ is given by

$$M^i(t) = \exp(-\alpha t)\alpha^{-1}\sum_{n=0}^{\infty}\frac{(\alpha t)^{n+1}}{(n+1)!}\sum_{j=0}^{n}B^j(e_ie_i^{\mathrm{T}})B^{n-j}.$$

with e_i^{T} denoting the transpose of the unit vector e_i. However, this expansion is fairly time consuming and for high dimensional matrices intractable. Moreover, the infinite sum has to be cut off at a finite n which entails inaccuracies. In the next subsection, we show how the left hand sides in (20) can be computed in an more efficient way. To end this subsection we finally state the resulting EM-algorithm 2 due to [19,1].

Algorithm 2 MLE-method according to [19,1]

Input: **Time series** $Y = \{y_0 = X(t_0),\ldots,y_N = X(t_N)\}$, initial guess of generator L_0.

Output: **MLE** $\widetilde{L}$.

(1) **Set** $\widetilde{L} := L_0$.

(2) **E-step: Compute for** $i,j,l,k = 1,\ldots,d$ **the conditional expectations**

$\mathbb{E}_{\widetilde{L}}[R_i(\tau)|X(\tau)=l,X(0)=k]$,

$\mathbb{E}_{\widetilde{L}}[N_{ij}(\tau)|X(\tau)=l,X(0)=k], i\neq j$ **via (22), (20)**

and

$\mathbb{E}_{\widetilde{L}}[R_i(T)|Y]$ **and** $\mathbb{E}_{\widetilde{L}}[N_{ij}(T)|Y]$ **via (19).**

(3) **M-Step: Setup the next MLE** $\widetilde{L}$ **of the generator by**

$$\tilde{l}_{ij} = \begin{cases}\mathbb{E}_{\widetilde{L}}[N_{ij}(T)|Y]/\mathbb{E}_{\widetilde{L}}[R_i(T)|Y], & i\neq j\\ -\sum_{k\neq i}\tilde{l}_{ik}, & \text{otherwise.}\end{cases}$$

(4) **Go to Step (2).**

4.3. Enhanced computation of the maximum likelihood estimator

It was shown in [4] that the conditional expectations $\mathbb{E}_L[N_{ij}(t)|X(t)=l,X(0)=k]$ and $\mathbb{E}_L[R_i(t)|X(t)=l, X(0)=k]$ can analytically be expressed in terms of the generator L. Recalling the notation of the transition matrix $P(s)=\exp(sL)$, the following identities have been proved

$$\begin{aligned}\mathbb{E}_L[R_i(t)|X(t)=l,X(0)=k] &= \frac{1}{p_{kl}(t)}\int_0^t p_{ki}(s)p_{il}(t-s)\mathrm{d}s,\\ \mathbb{E}_L[N_{ij}(t)|X(t)=l,X(0)=k] &= \frac{l_{ij}}{p_{kl}(t)}\int_0^t p_{ki}(s)p_{jl}(t-s)\mathrm{d}s.\end{aligned} \tag{24}$$

In [18], the Holmes and Rubin derived explicit formulas for the integrals in (24) by using the eigendecomposition

$$L = UD_\lambda U^{-1} \tag{25}$$

of the generator L. Here, the columns of the matrix U consist of all eigenvectors to the corresponding eigenvalues of L in the diagonal matrix $D_\lambda = \mathrm{diag}(\lambda_1,\ldots,\lambda_d)$. Consequently, the expression of the transition matrix $P(t)$ simplifies to

$$P(t) = \exp(tL) = U\exp(tD_\lambda)U^{-1}$$

and we finally end up with a closed form expression of the integrals in (24), that is

$$\int_0^t p_{ab}(s)p_{cd}(t-s)\mathrm{d}s = \sum_{p=1}^{d} u_{ap}u_{pb}^{-1}\sum_{q=1}^{d} u_{cq}u_{qd}^{-1}\Psi_{pq}(t) \tag{26}$$

where the symmetric matrix $\Psi(t)=(\Psi_{pq}(t))_{p,q\in S}$ is defined as

$$\Psi_{pq}(t) = \begin{cases} te^{t\lambda_p} & \text{if } \lambda_p=\lambda_q \\ \frac{e^{t\lambda_p}-e^{t\lambda_q}}{\lambda_p-\lambda_q} & \text{if } \lambda_p\neq\lambda_q. \end{cases} \tag{27}$$

In many cases, computing the conditional expectations explicitly via the eigendecomposition leads to a considerable speed-up of the method. Therefore, this alternative will be referred to as the enhanced MLE-method. For the convenience of the reader, this method is summarized in Algorithm 3. In a single iteration step, d^2 conditional expectations have to be computed where each one is decomposed into d^2 conditional expectations. Hence, the computational cost of a single iteration step in the Algorithms 2 and 3 is $\mathcal{O}(d^4\cdot T_{\mathbb{E}})$ where $T_{\mathbb{E}}$ denotes the computational cost to compute a single conditional expectation in the E-Step. In many applications, the frequency matrix C is sparse, i.e., if $|C| = |\{c_{ij}: c_{ij}>0,\, i,j\in S\}|$ denotes the number of positive entries in C then $|C|\ll d^2$. In this case the computational cost in both algorithms for a single iteration reduces to $\mathcal{O}(|C|^2\cdot T_{\mathbb{E}})$. The numerical considerations in [1] lead to a total computational cost per iteration in Algorithm 2 of at least $\mathcal{O}(d^6)$. According to the closed form expressions for the expectations (26), the computational cost of a single iteration in the enhanced MLE-method (Algorithm 3) is $\mathcal{O}(d^5)$ which is achieved by a simultaneously computation of the unknowns via matrix multiplication. For example, define for a fixed $i\in S$ the matrix $M^i_{kl} = \mathbb{E}_L[R_i(\tau)|X(\tau)=l,X(0)=k]$. Let U_i^{-1} denote the i^{th} row of the matrix U^{-1} and U_i the i^{th} column of U. Then M^i can be computed by

$$M^i = U[(U_i^{-1}U_i)*\Psi]U^{-1}$$

where $A*B$ is the Hadamard (entrywise) product of two matrices A and B.

Algorithm 3 Enhanced MLE-method

Input: **Time series** $Y = \{y_0 = X(t_0), \ldots, y_N = X(t_N)\}$, initial guess of generator L_0.
Output: **MLE** $\widetilde{L}$.
(1) **Set** $\widetilde{L} := L_0$.
(2) **Compute eigendecomposition (25) of** $\widetilde{L}$.
(3) **E-step: Compute for** $i,j,l,k = 1,\ldots,d$ **the conditional expectations**
$\mathbb{E}_{\widetilde{L}}[R_i(\tau)|X(\tau) = l, X(0) = k]$,
$\mathbb{E}_{\widetilde{L}}[N_{ij}(\tau)|X(\tau) = l, X(0) = k], i \neq j$ **via (26),(24)**
and
$\mathbb{E}_{\widetilde{L}}[R_i(T)|Y]$,
$\mathbb{E}_{\widetilde{L}}[N_{ij}(T)|Y]$ **via (19).**
(4) **M-Step: Setup the next MLE** $\widetilde{L}$ **of the generator by**

$$\tilde{l}_{ij} = \begin{cases} \mathbb{E}_{\widetilde{L}}[N_{ij}(T)|Y]/\mathbb{E}_{\widetilde{L}}[R_i(T)|Y], & i \neq j \\ -\sum_{k\neq i}\tilde{l}_{ik}, & \text{otherwise.} \end{cases}$$

(5) **Go to Step (2).**

4.4. Reversible case

In the reversible case the homogeneous Markov jump process, given by its generator L, admits an unique stationary distribution $\pi = (\pi_i)_{i\in S}$ and, moreover, *detailed balance* holds:

$$l_{ji} = \frac{\pi_i}{\pi_j} l_{ij}.$$

This has two important consequences for the EM-algorithm. The first one is that detailed balance guarantees a special representation of L which improves the stability and accuracy of the EM-algorithm. Furthermore, one has to take into account that the M-step in general does not preserve the reversibility. To understand the first issue, notice that L can be written as

$$L = D_\pi^{-1/2}\mathbf{S}D_\pi^{1/2}$$

with a symmetric matrix $\mathbf{S}$ which can be decomposed as

$$\mathbf{S} = VD_\lambda V^{\mathrm{T}}$$

where $\lambda_1, \ldots, \lambda_d \in \mathbb{R}$ are the eigenvalues of $\mathbf{S}$ and V is an orthogonal matrix, i.e. $VV^{\mathrm{T}} = I$. Combining things, we end up with [4]

$$P(t) = D_\pi^{-1/2} V \exp(D_\lambda) V^{\mathrm{T}} D_\pi^{1/2}$$

where $D_\pi^{1/2} = \mathrm{diag}(\sqrt{\pi_1}, \ldots, \sqrt{\pi_d})$. Consequently, the integrals in (24) reduce to [18]

$$\int_0^t p_{ab}(s)p_{cd}(t-s)\mathrm{d}s = \left(\frac{\pi_b\pi_d}{\pi_a\pi_c}\right)^{1/2} \sum_{p=1}^{d} v_{ap}v_{bp} \sum_{q=1}^{d} v_{cq}v_{dq}\Psi_{pq}(t) \tag{28}$$

where Ψ is defined in (27). Next, we turn our attention to the problem of the non-preservation of the reversibility in the M-Step. The first idea could be to exploit the fact that detailed balance implies the bisection of the unknowns because l_{ji} is determined by π_i, π_j and l_{ij}. Then one could proceed as follows: firstly, compute the MLE $\widetilde{L}$ via the EM-algorithm as usual and then define a reversible generator $\widetilde{L}^{\mathrm{R}} = (\tilde{l}_{ij}^{\mathrm{R}})_{i,j\in S}$ by

$$\tilde{l}_{ij}^{\mathrm{R}} = \begin{cases} \tilde{l}_{ij} & \text{if } i \leqslant j \\ \frac{\pi_j}{\pi_i}\tilde{l}_{ji} & \text{otherwise.} \end{cases}$$

This would work in principle but it does not guarantee that the resulting generator $\widetilde{L}^{\mathrm{R}}$ is the MLE *subject* to the space of reversible generators. As a remedy, we include the restriction to that space explicitly in the log-likelihood function (17) via Lagrange multiplier:

$$\mathcal{G}^{\mathrm{R}}(L;L_0) = \mathcal{G}(L;L_0) + \sum_{i=1}^{d}\sum_{j>i}^{d} \mu_{ij}(\pi_i l_{ij} - \pi_j l_{ji}).$$

Performing the usual steps, we end up with the MLE $\widetilde{L}^{\mathrm{R}}$, given by

$$\tilde{l}_{ij}^{\mathrm{R}} = \begin{cases} \dfrac{\mathbb{E}_{L_0}\left[N_{ij}(T)|Y\right]}{-\mu_{ij}\pi_i + \mathbb{E}_{L_0}[R_i(T)|Y]}, & i<j \\ \dfrac{\pi_i}{\pi_j}\tilde{l}_{ij}^{\mathrm{R}}, & \text{otherwise} \end{cases} \tag{29}$$

where the Lagrange multiplier can be determined by

$$\mu_{ij} = \left[\frac{\mathbb{E}_{L_0}[R_j(T)|Y]}{\pi_j \mathbb{E}_{L_0}[N_{ji}(T)|Y]} - \frac{\mathbb{E}_{L_0}[R_i(T)|Y]}{\pi_i \mathbb{E}_{L_0}[N_{ij}(T)|Y]}\right] \times \left[-\frac{\mathbb{E}_{L_0}[N_{ij}(T)|Y]\cdot \mathbb{E}_{L_0}[N_{ji}(T)|Y]}{\mathbb{E}_{L_0}[N_{ij}(T)|Y] + \mathbb{E}_{L_0}[N_{ji}(T)|Y]}\right]. \tag{30}$$

Combining both issues leads to Algorithm 4.

Algorithm 4 Enhanced MLE-method for the reversible case

Input: **Time series** $Y = \{y_0 = X(t_0), \ldots, y_N = X(t_N)\}$, initial guess of reversible generator L_0^{R}.
Output: **Estimated generator** $\widetilde{L}^{\mathbf{R}}$.
(1) **Set** $\widetilde{L}^{\mathbf{R}} := L_0^{\mathbf{R}}$.
(2) **Compute eigendecomposition of** $\widetilde{L}^{\mathbf{R}}$.
(3) **E-step: Compute the conditional expectations**
$\mathbb{E}_{\widetilde{L}^{\mathbf{R}}}[R_i(\tau)|X(\tau)=l, X(0)=k]$,
$\mathbb{E}_{\widetilde{L}^{\mathbf{R}}}[N_{ij}(\tau)|X(\tau)=l, X(0)=k]$ **via (28),(24)**
and
$\mathbb{E}_{\widetilde{L}^{\mathbf{R}}}[R_i(T)|Y]$,
$\mathbb{E}_{\widetilde{L}^{\mathbf{R}}}[N_{ij}(T)|Y]$ **via (19)**
(4) **Compute Lagrange multipliers** μ_{ij} **via (30)**
(5) **M-Step: Setup the next MLE** $\widetilde{L}^{\mathbf{R}}$ **of the generator via (29).**
(6) **Go to Step (2).**

4.5. Scaling

We prove that the maximizer (18) in the (enhanced) MLE-method respects the time invariance of the semigroup $P(t) = \exp(tL)$. Consequently, we can estimate a generator $\widetilde{L}(\tau')$ with respect to $\tau' = 1$ and regain the original generator by $\widetilde{L}(\tau) = \widetilde{L}(1)/\tau$.

Lemma 1. *Let $\widetilde{L}(\tau)$ be the MLE with respect to the time lag τ and $\widetilde{L}(1)$ with respect to $\tau' = 1$. Then for both cases the general and the reversible case the following relation holds*:

$$\widetilde{L}(\tau) = \frac{1}{\tau}\widetilde{L}(1). \tag{31}$$

Proof. A short calculation shows that

$$\int_0^\tau p_{ab}(s)p_{cd}(\tau - s)\mathrm{d}s = \tau \int_0^1 [\exp(s\overline{L})_{ab}(\exp((1-s)\overline{L})_{cd}]\mathrm{d}s$$

where $\overline{L} = \tau L$. But this immediately implies

$$\mathbb{E}_L[R_i(\tau)|X(\tau)=l, X(0)=k] = \tau \mathbb{E}_{\overline{L}}[R_i(1)|X(1)=l, X(0)=k]$$

and, by noting that $l_{ij} = \frac{1}{\tau}\overline{l}_{ij}$,

$$\mathbb{E}_L[N_{ij}(\tau)|X(\tau)=l, X(0)=k] = \mathbb{E}_{\overline{L}}[N_{ij}(1)|X(1)=l, X(0)=k]$$

which proves (31). In the reversible case the same reasoning shows that the Lagrange multipliers scale linearly with τ and therefore (31) also holds. □

4.6. Enhanced MLE-method vs. MLE-method

The eigendecomposition approach has several advantages compared to the numerical considerations proposed in [1]. Let d be the dimension of the discrete state space. As explained in Section 4.2, the computational cost is reduced to $\mathcal{O}(d^5)$ thanks to the closed form expression (26). Moreover, there is no longer an explicit dependency on the length of the time series. The second advantage is the exact computation of the conditional expectations involved in the E-step of the EM-algorithm. The steps which introduce numerical errors are the eigendecomposition and the computation of U^{-1}. As before, the explicit inversion of U can be avoided by considering the left eigenvectors of $\widetilde{L}$. We are aware that the eigendecomposition of non-symmetric matrices can be *ill-conditioned*, but any reliable numerical solver should indicate this. Nevertheless, the computational cost of both steps ($\mathcal{O}(d^3)$) and their numerical stability are superior compared to any numerical approximation scheme for solving the ODEs in (22).

5. Numerical examples

5.1. Preparatory considerations

In order to compare the performance of the quadratic programming approach (QP) and the maximum likelihood method (MLE), these approaches are now applied to a series of model problems. A comparison with the resolvent method is omitted because, as we have seen above, this method does not respect the generator constraints and produces invalid estimates $\widetilde{L} \notin \mathfrak{G}$ when no generator exists. A rather straightforward test would proceed as follows:

(1) Choose an arbitrary generator $L \in \mathfrak{G}$ and a time lag τ.
(2) Compute the corresponding transition matrix $P(\tau) = \exp(\tau L)$.
(3) Produce a time series $Y = \{y_0 = X(t_0), \ldots, y_N = X(t_N)\}$ by sampling from $P(\tau)$.
(4) Pass this data to each of the two methods and compute an estimate $\widetilde{L} \approx L$.
(5) Compare the errors of the two approaches.

Although such a test seems to be somewhat reasonable, we will *not* use this procedure. The reason for our refusal is the fact that the time series produced in step 3 is just a single realization. Hence, the result of this test is random, too, and applying the test several times to the methods yields different results even though the input L remains unchanged. In fact, both methods are affected by the sampling error

$$\|P(\tau) - \widehat{P}\| \quad \text{with } \hat{P} = (\hat{p}_{ij})_{i,j} \quad \text{and} \quad \hat{p}_{ij} = \frac{c_{ij}}{\sum_{j=1}^{d} c_{ij}}. \tag{32}$$

(Here and below, $\|\cdot\|$ denotes the matrix 2-norm.) Roughly speaking, the sampling error indicates how well the frequency matrix of a time series "represents" the underlying transition matrix. In the limit $N \to \infty$ one may expect the sampling error to vanish, but for a finite number of observations the deviation can be considerable. Since the outcome of a numerical method cannot be better than the input data, the error of both methods are bounded from below by the sampling error.

Therefore, our numerical experiments are designed in a different way:

(1) (a) Choose a generator $L \in \mathfrak{G}$ and a time lag τ and compute the corresponding transition matrix $P(\tau) = \exp(\tau L)$,

or

(b) choose a transition matrix P. This allows to test the performance of the methods in situations where no underlying generator exists. In this case, the time lag does not matter, and we can set $\tau = 1$.

(2) Define a virtual frequency matrix by multiplying each row of the transition matrix $P(\tau)$ with the corresponding entry of the stationary distribution $\pi = (\pi_i), i \in S$ and the length N of the (virtual) time series:

$$c_{ij} = \text{round}(N \pi_i p_{ij}). \tag{33}$$

This is the frequency matrix which, up to rounding errors, reflects the underlying transition matrix in an optimal way.

(3) Based on the virtual frequency matrix, define the virtual transition matrix

$$\widehat{P}_{virt} = (\hat{p}_{ij})_{i,j} \quad \text{and } \hat{p}_{ij} = \frac{c_{ij}}{\sum_{j=1}^{d} c_{ij}} \tag{34}$$

and compute an estimate $\widetilde{L} \approx L$ for the generator.

(4) For both methods, compute and compare the errors:
   (a) $\|\widetilde{L} - L\|$ (only if L is available, i.e. if variant (a) of step 1 was used).
   (b) $\|P(\tau) - \exp(\tau\widetilde{L})\|$.
   (c) $\|\widehat{P}_{\text{virt}} - \exp(\tau\widetilde{L})\|$ with $\widehat{P}_{\text{virt}}$ defined in (34).

The advantage of this approach to numerical experiments is illustrated by a simple example in Appendix A, cf. Section A.2.

Of course, the choice of the initial value L_0 for the MLE-method is crucial for the convergence. If the matrix logarithm of $\widehat{P}_{\text{virt}}$ exists, then a good initial value L_0 can easily be obtained by taking the absolute values of the off-diagonal entries of $\log(\widehat{P}_{\text{virt}})/\tau$ and setting the diagonal entries to the corresponding negative row sums, respectively.

5.2. Transition matrix with underlying generator

In a first example we follow variant (a) of step 1 and consider the generator

$$L = \begin{pmatrix}
-4.293 & 0.678 & 0.301 & 0.819 & 0.592 & 0.149 & 0.543 & 0.411 & 0.774 & 0.023 \\
0.033 & -3.833 & 0.633 & 0.260 & 0.636 & 0.878 & 0.485 & 0.527 & 0.147 & 0.231 \\
0.857 & 0.995 & -5.466 & 0.704 & 0.532 & 0.021 & 0.441 & 0.920 & 0.148 & 0.845 \\
0.682 & 0.499 & 0.005 & -4.691 & 0.208 & 0.923 & 0.626 & 0.379 & 0.639 & 0.726 \\
0.801 & 0.430 & 0.816 & 0.082 & -4.268 & 0.632 & 0.077 & 0.638 & 0.093 & 0.694 \\
0.917 & 0.829 & 0.690 & 0.875 & 0.241 & -5.584 & 0.544 & 0.173 & 0.928 & 0.383 \\
0.388 & 0.116 & 0.981 & 0.077 & 0.720 & 0.632 & -4.667 & 0.785 & 0.485 & 0.479 \\
0.472 & 0.598 & 0.069 & 0.741 & 0.400 & 0.753 & 0.270 & -4.435 & 0.163 & 0.967 \\
0.088 & 0.221 & 0.045 & 0.125 & 0.394 & 0.769 & 0.291 & 0.776 & -3.495 & 0.783 \\
0.925 & 0.398 & 0.740 & 0.443 & 0.411 & 0.808 & 0.822 & 0.342 & 0.131 & -5.022
\end{pmatrix} \in \mathfrak{G}. \tag{35}$$

Based on the exact transition matrix $P(\tau)$ with $\tau = 0.2$, we computed the virtual transition matrix $\widehat{P}_{\text{virt}}, N = 10^{10}$ via (34) and estimated the generator with both methods. The enhanced MLE-method (3) stopped after 1132 iteration steps because the increment-based stopping criterion $\|\widetilde{L}_k - \widetilde{L}_{k-1}\| \leqslant \text{tol}$ with $\text{tol} = 10^{-7}$ had been met. Fig. 1 shows the error of $\widetilde{L}_{\text{MLE}}$ with respect to L (35) as a function of the iteration steps.

Obviously, the convergence of the enhanced MLE-method is very slow. In contrast to the MLE-method, the QP-method converged after only a few iteration steps. In Table 1, the errors of both approaches are compared. The QP-approach approximates the original generator clearly better than the enhanced MLE-method. This is, however, not surprising because it has to be taken into account that the QP-approach approximates the eigendecomposition of $\widehat{P}_{\text{virt}}$ and for the length $N = 10^{10}$ of a virtual time series the difference between the exact and the virtual transition matrix is only $\|P(0.2) - \widehat{P}_{\text{virt}}\| = 1.39 \cdot 10^{-9}$.

Next, we investigate the influence of the sampling error on both estimation methods. Instead of considering realizations of the Markov jump process, we compute estimations of L for a number of virtual time series of increasing length N. Fig. 2 shows the resulting errors of $\widetilde{L}_{\text{MLE}}$ and $\widetilde{L}_{\text{QP}}$ with respect to the generator L (35) as a

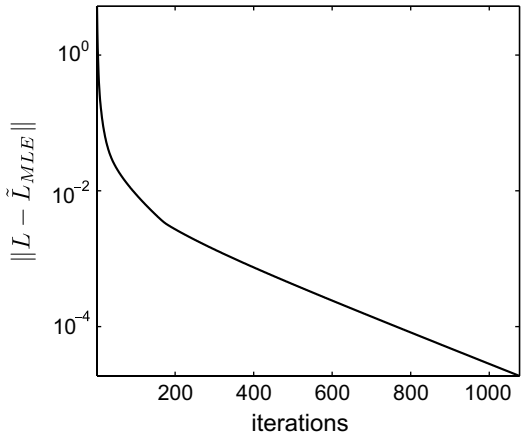

Fig. 1. Approximation error of $\widetilde{L}_{\mathrm{MLE}}$ with respect to the generator L (35) as a function of the iteration steps.

Table 1
Approximation errors of the estimated generators $\widetilde{L}_{\mathrm{QP}}$ and $\widetilde{L}_{\mathrm{MLE}}$ with respect to the given generator (35), the exact transition matrix $P(\tau)$ and the transition matrix $\widehat{P}_{\mathrm{virt}}$ constructed via (34)

	$\|L - \widetilde{L}\|$	$\|\exp(\tau L) - \exp(\tau\widetilde{L})\|$	$\|\widehat{P}_{\mathrm{virt}} - \exp(\tau\widetilde{L})\|$
QP	2.07×10^{-8}	1.39×10^{-9}	1.18×10^{-14}
MLE	1.88×10^{-5}	1.19×10^{-6}	1.19×10^{-6}

Results for the time lag $\tau = 0.2$ and the length of the virtual time series $N = 10^{10}$.

function of the length N of the virtual time series. It reveals that for a time series of a realistic length ($N \leqslant 10^7$), the errors of $\widetilde{L}_{\mathrm{QP}}$ and $\widetilde{L}_{\mathrm{MLE}}$ are almost identical. The fact that the error of the MLE-method remains larger than 10^{-5} regardless of N is due to the chosen stopping criterion.

5.3. Transition matrix without underlying generator

In contrast to the first case both estimation procedures are now applied to a transition matrix which does not possess a generator:

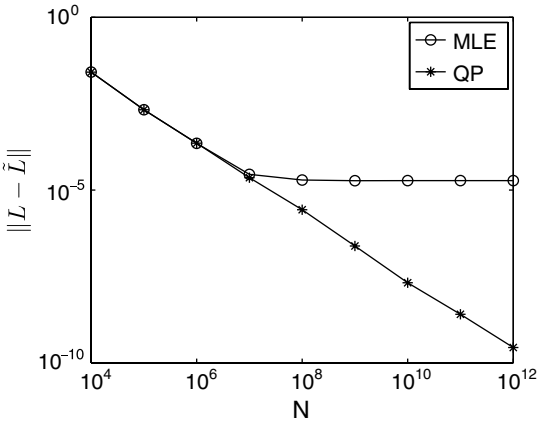

Fig. 2. Graphs of the errors of $\widetilde{L}_{\mathrm{MLE}}$ and $\widetilde{L}_{\mathrm{QP}}$ with respect to the generator L (35), respectively, as a function of the length N of the virtual time series. The error of the MLE-method remains larger than 10^{-5} regardless of N due to the stopping criterion $\|\widetilde{L}_k - \widetilde{L}_{k-1}\| \leqslant 10^{-7}$.

$$P = \begin{pmatrix} 0.6455 & 0.0376 & 0.0338 & 0.0394 & 0.0467 & 0.0626 & 0.0406 & 0.0032 & 0.0316 & 0.0591 \\ 0.0146 & 0.7924 & 0.0549 & 0.06 & 0.0103 & 0 & 0 & 0 & 0.0162 & 0.0516 \\ 0.0497 & 0.0656 & 0.7516 & 0.0698 & 0.0009 & 0 & 0 & 0 & 0.0469 & 0.0155 \\ 0.0208 & 0.0565 & 0.0577 & 0.7238 & 0.0615 & 0 & 0 & 0 & 0.022 & 0.0577 \\ 0.0376 & 0.0447 & 0.0394 & 0.061 & 0.7072 & 0 & 0 & 0 & 0.0666 & 0.0436 \\ 0.0105 & 0.0571 & 0.0258 & 0.0121 & 0.0208 & 0.7279 & 0.0322 & 0.0536 & 0.0507 & 0.0093 \\ 0 & 0 & 0 & 0.0699 & 0.0472 & 0.0161 & 0.7535 & 0.0692 & 0.0294 & 0.0148 \\ 0 & 0 & 0 & 0.019 & 0.0199 & 0.0406 & 0.0556 & 0.7701 & 0.0522 & 0.0425 \\ 0 & 0 & 0 & 0.0191 & 0.0355 & 0.0575 & 0.0045 & 0.0596 & 0.7762 & 0.0476 \\ 0 & 0 & 0 & 0.0657 & 0.0049 & 0.0398 & 0.0453 & 0.0329 & 0.033 & 0.7784 \end{pmatrix} \notin \mathfrak{P} \tag{36}$$

One can immediately verify via Theorem 2 cited in the Appendix that (36) cannot be generated since, e.g, the state 6 is accessible from state 2 via state 1 but $p_{2,6} = 0$. As Table 2 shows, the errors of the estimated transition matrices $\exp(\tau\widetilde{L})$ are of the same order of magnitude and are larger than in the first example due to the additional difficulty that no generator exists.

The error $\|P - \exp(\widetilde{L}_{\mathrm{MLE}})\|$ as a function of the first 10 iteration steps is shown in the upper panel of Fig. 3. Surprisingly, the best accuracy is obtained after only one iteration, but the following iterations increase the error again. The reason for this behavior is the fact that the MLE-method aims to maximizing the likelihood instead of minimizing the error, and the graph of the discrete log-likelihood, depicted in the lower panel of Fig. 3, clearly shows that the maximum likelihood was not attained after the first iteration.

In contrast to the first example, Fig. 4 shows that here increasing the length of the virtual time series does not improve the estimation significantly in both methods.

5.4. Transition matrix with exact generator under perturbation

In the next example, we consider again the transition matrix $P(\tau)$ with $\tau = 0.2$ which is generated by the generator (35) given in the first example. In order to investigate the impact of perturbations due to, e.g., sampling from a time series, we estimate a generator based on a perturbed transition matrix

$$P_\epsilon(\tau) = \exp(\tau L) + k\epsilon, \quad k = 0, \ldots, 19,$$

where ϵ is the perturbation matrix

$$\epsilon = 10^{-5} \cdot \begin{pmatrix} 4.055 & -3.552 & 1.754 & 0.805 & -4.090 & -3.519 & 4.719 & 0.047 & 0.696 & -0.917 \\ 3.104 & -3.508 & -1.609 & 2.874 & 1.319 & -0.671 & 2.020 & 1.459 & 1.272 & -6.261 \\ -3.22 & -0.978 & -2.611 & 5.673 & -3.653 & 2.386 & 5.726 & -2.478 & 0.154 & -0.993 \\ 4.467 & -1.238 & -5.225 & 1.944 & -1.021 & -3.496 & 2.433 & -2.047 & 2.687 & 1.497 \\ 4.698 & -4.188 & -1.271 & 1.949 & -4.191 & -0.450 & -0.850 & 3.649 & -4.336 & 4.991 \\ 4.376 & -2.336 & -1.603 & 3.415 & 1.556 & 1.850 & -4.529 & -2.277 & 4.355 & -4.808 \\ 1.200 & -2.234 & 5.509 & -4.121 & -1.151 & -0.133 & -3.341 & -3.631 & 4.118 & 3.785 \\ 2.836 & -1.009 & 2.731 & -3.009 & -1.067 & -4.559 & 2.699 & 2.614 & 3.194 & -4.432 \\ -1.478 & 4.040 & -0.318 & -3.722 & -0.412 & 1.249 & 0.450 & -2.992 & -2.153 & 5.336 \\ -1.460 & -1.569 & 5.235 & -0.772 & -2.618 & 4.252 & -2.006 & -0.251 & 0.705 & -1.514 \end{pmatrix}.$$

Table 2
Approximation errors of $\exp(\widetilde{L}_{\mathrm{QP}})$ and $\exp(\widetilde{L}_{\mathrm{MLE}})$ with respect to the given transition matrix (36) and the transition matrix $\widehat{P}_{\mathrm{virt}}$ constructed via (34)

	$\|P - \exp(\tau\widetilde{L})\|$	$\|\widehat{P}_{\mathrm{virt}} - \exp(\tau\widetilde{L})\|$
QP	1.74×10^{-2}	1.74×10^{-2}
MLE	2.86×10^{-2}	2.86×10^{-2}

Results of MLE-method for $\mathrm{tol} = 10^{-7}$.

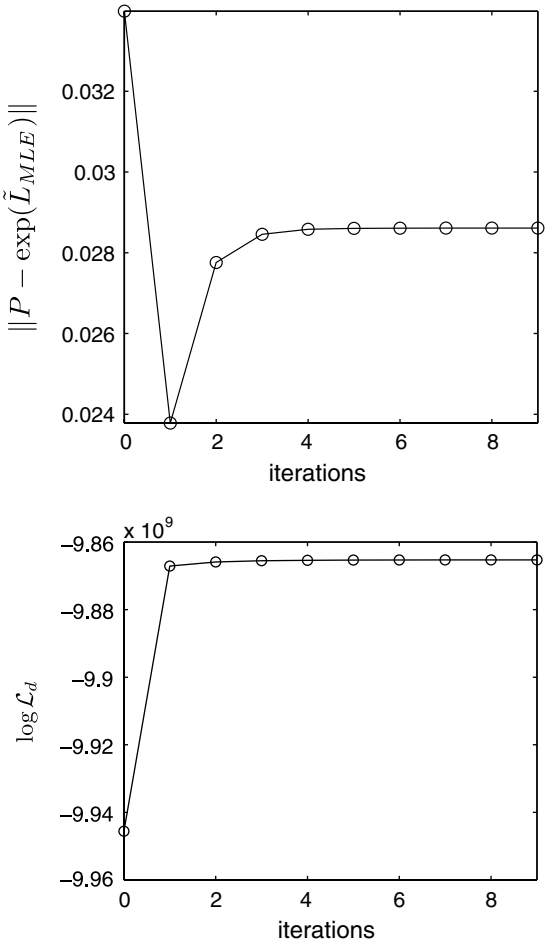

Fig. 3. Upper: error of $\exp(\widetilde{L}_{\mathrm{MLE}})$ with respect to the transition matrix P in (36) as a function of the first 10 iteration steps. Lower: the discrete log-likelihood $\mathcal{L}_d$ as a function of the 10 first iteration steps.

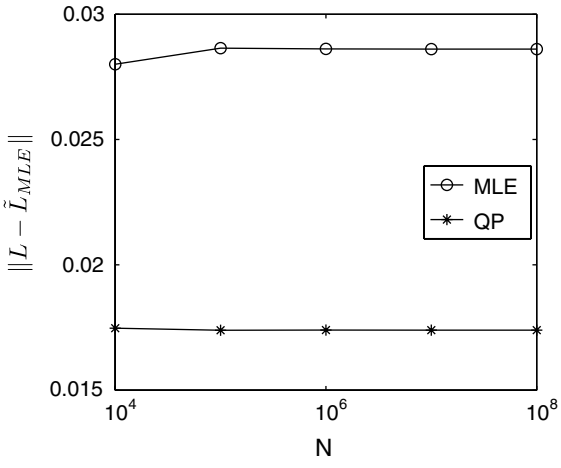

Fig. 4. The graphs of the errors of $\exp(\widetilde{L}_{\mathrm{MLE}})$ and $\exp(\widetilde{L}_{\mathrm{QP}})$ with respect to the transition matrix in (36), respectively, as a function of the length N of the virtual time series.

The upper panel of Fig. 5 shows the deviation of the estimated generators from the unperturbed generator as a function of the perturbation factor k. The QP-method performs slightly better but both errors $\|L - \widetilde{L}_{\mathrm{QP}}\|$ and $\|L - \widetilde{L}_{\mathrm{MLE}}\|$ are of the same order of magnitude. Furthermore, the errors scale linearly with the perturbation factor k. This observation is plausible since for small perturbations the logarithm $\log(P + \epsilon)$ can be approximated by $\log(P) + \mathcal{O}(\epsilon)$. The lower panel of Fig. 5 illustrates the behavior of the errors of the estimated transition matrices $\exp(\tau\widetilde{L}_{\mathrm{QP}})$ and $\exp(\tau\widetilde{L}_{\mathrm{MLE}})$, respectively. A similar reasoning as above explains the linear scaling.

Finally, we consider the error of the estimated transition matrices $\exp(\tau\widetilde{L}_{\mathrm{QP}})$ and $\exp(\tau\widetilde{L}_{\mathrm{MLE}})$ with respect to the perturbed transition matrix $P_\epsilon(\tau) = \exp(\tau L) + k\epsilon$, depicted in Fig. 6. Notice that the error $\|P_\epsilon(\tau) - \exp(\tau\widetilde{L})\|$ is bounded from above, namely

$$\|P_\epsilon(\tau) - \exp(\tau\widetilde{L})\| \leqslant \|\exp(\tau L) - \exp(\tau\widetilde{L})\| + k\|\epsilon\|.$$

Indeed, Fig. 6 shows that both errors obey that bound. For the perturbation factors up to $k = 8$, the matrix logarithm of P_ϵ is still a generator whereas for $k = 9, \ldots, 19$ the perturbation is apparently high enough to destroy the generator structure of the matrix logarithm of P_ϵ. However, the accuracy of both methods is again of the same order of magnitude.

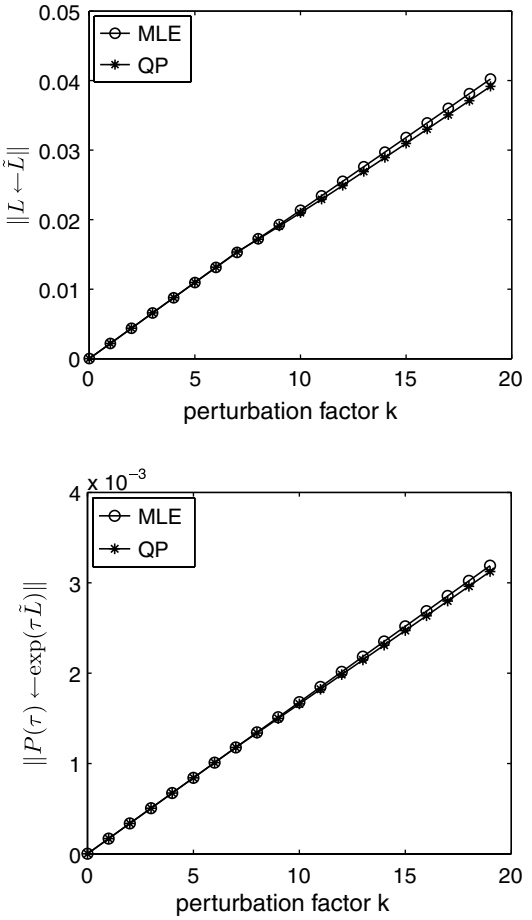

Fig. 5. Upper: approximation error of the generator estimates $\widetilde{L}_{\mathrm{QP}}$ and $\widetilde{L}_{\mathrm{MLE}}$ with respect to the unperturbed generator (35) as a function of the perturbation factor k. Lower: error of the estimated transition matrices $\exp(\tau\widetilde{L}_{\mathrm{QP}})$ and $\exp(\tau\widetilde{L}_{\mathrm{MLE}})$ with respect to the unperturbed transition matrix $\exp(\tau L)$ as a function of the perturbation factor k. Results for $\tau = 0.2$.

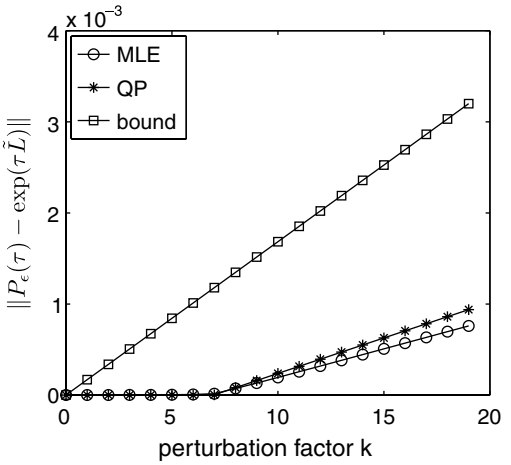

Fig. 6. Error of the estimated transition matrices $\exp(\tau\widetilde{L}_{\mathrm{QP}})$ and $\exp(\tau\widetilde{L}_{\mathrm{MLE}})$ with respect to the perturbed transition matrix $P_\epsilon(\tau) = \exp(\tau L) + k\epsilon$ as a function of the perturbation factor k. The upper bound was computed via $\widetilde{L}_{\mathrm{MLE}}$.

5.5. Application to a time series from molecular dynamics

In the last example, we apply both methods to a time series of two torsion angles extracted from a numerical simulation of a trialanine dipeptide analogue. The ball-and-stick representation of this molecule together with the two considered torsion angles Φ and Ψ is shown in the upper panel of Fig. 7. The considered time series was generated in vacuum at a temperature of 750 K using the Hybrid Monte Carlo method [8] with 544.500 steps and with the GROMACS force field [9,10]. The integration of the subtrajectories of the Monte Carlo proposal step were realized with 100 1-fs time steps of the Verlet integration scheme which results in the time lag $\tau = 10^{-13}$.

We are interested in identifying conformations of the trialanine molecule. A conformation of a molecule is understood as a mean geometric structure of the molecule which is conserved on a large time scale compared to the fastest molecular motions. From the dynamical point of view, a conformation typically persists for a long time (again compared to the fastest molecular motions) such that the associated subset of configurations is *metastable*. The lower panel of Fig. 7 shows the projection of the time series onto the torsion angles Φ and Ψ which reveals the metastable behavior. The Ramachandran plot of the time series, given in the upper panel of Fig. 8, illustrates the dependency among both torsion angles and indicates that the trialanine molecule attains three different conformations.

The metastability analysis via the *transfer operator approach* is based on a box-discretization of the torsion angle space. Identifying each element of the given time series with the box by which it is covered, we built up a discrete transition matrix P by counting the transitions between the boxes. The enhanced properties of the transition matrix P, namely the almost constant levels in the dominant eigenvectors of P, allows to identify conformations via Perron cluster cluster analysis (PCCA) [11,12].

To gain further insight into the dynamics of a molecule, the transitions between its conformations and the corresponding rates have to be considered. A recently developed framework, the Transition Path Theory (TPT), provides a statistical theory to describe transitions in Markov processes [13,14] and allows to compute, e.g., transitions rates between conformations. The basic object in TPT is the committor function q_{AB} with respect to two disjoint subsets A and B of the state space. Let $i \in S$ denote a state, then $q_{AB}(i)$ is the probability to go rather to B than to A conditioned on the process has started in i. For a Markov jump process, it is shown in [15] that the committor $q_{AB}(i), i \in S$ satisfies

$$\begin{cases} (Lq_{AB})(i) = 0, i \in S \setminus (A \cup B), \\ q_{AB}(i) = 0, i \in A, \\ q_{AB}(i) = 1, i \in B, \end{cases} \tag{37}$$

where L is the generator of the Markov jump process.

$\leq 0.5\}$

boxes) where q_{AB} is the conditional func-

as the box in which the stationary distribution restricted on

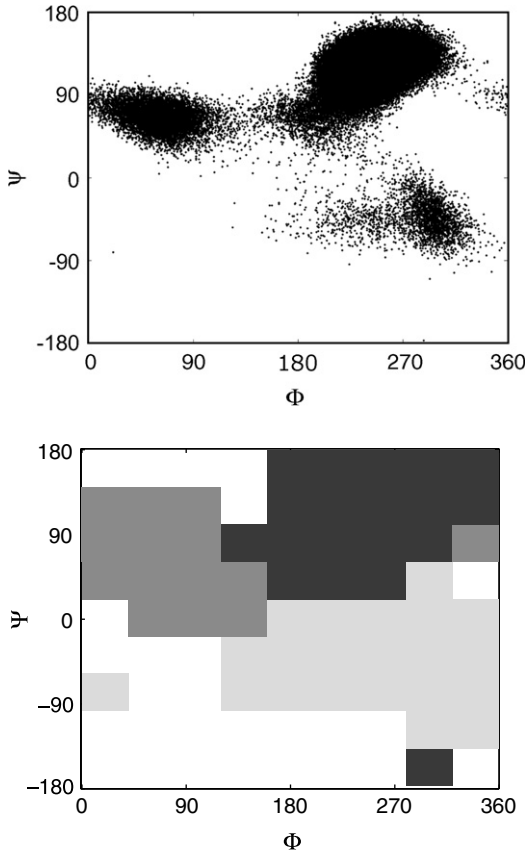

Fig. 8. Upper: Ramachandran plot of the torsion angles. Lower: Decomposition of the torsion angle state space into three metastable sets via PCCA. Results for an equidistant discretization of the torsion angle space into 9×9 boxes.

Table 3
The first four largest eigenvalues of the transition matrix $\widehat{P}$ and the transition matrices computed from the estimated generator $\widetilde{L}_{\mathrm{QP}}$ and $\widetilde{L}_{\mathrm{MLE}}$

	λ_1	λ_2	λ_3	λ_4
$\widehat{P}(\tau)$	1	0.9932	0.9928	0.7560
$\exp(\tau\widetilde{L}_{\mathrm{QP}})$	1	0.9918	0.9913	0.7231
$\exp(\tau\widetilde{L}_{\mathrm{MLE}})$	1	0.9929	0.9920	0.7290

The gap between the third and fourth eigenvalue suggests a decomposition of the torsion angle space into three metastable sets. Results for a 9×9 box-discretization.

Table 4
The four largest eigenvalues of the transition matrix $\hat{P}(\tau)$ and the transition matrices computed from the estimated generator $\widetilde{L}_{\mathrm{MLE}}$

	λ_1	λ_2	λ_3	λ_4
$\hat{P}(\tau)$	1	0.9955	0.9952	0.8543
$\exp(\tau\widetilde{L}_{\mathrm{MLE}})$	1	0.9951	0.9949	0.8440

Results for an equidistant 20×20 box-discretization of the torsion angle space.

the lower conformation attains its maximum. The set B was chosen analogously with respect to the upper conformation. As one can see, the committor sharply separates the lower conformation from the upper conformations.

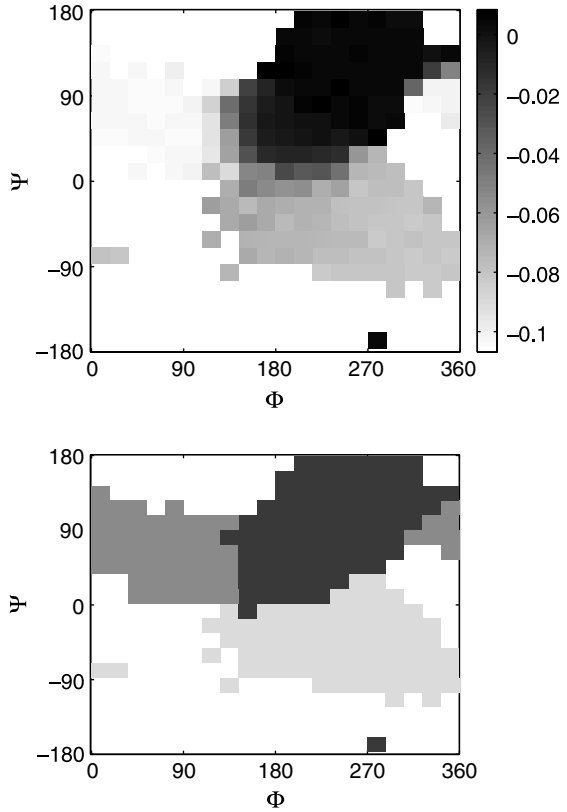

Fig. 9. Upper: box plot of the right eigenvector corresponding to the second largest eigenvalue of L_{MLE}. The three almost constant levels indicate the three metastable sets. Lower: decomposition of the torsion angle state space via PCCA into three metastable sets. Results for an equidistant 20×20 box-discretization of the torsion angle space.

6. Discussion and summary

To sum up, we discuss the pros and cons of the QP-method and the enhanced MLE-method. Concerning the accuracy of the estimates, the extensive numerical tests presented in the previous section did not reveal any decisive difference between both approaches. Only in case of an extremely long time series produced by a transition matrix with underlying generator, the QP-method produced significantly better results than the MLE-method. This case, however, can be considered as somewhat artificial because a time series with such a huge number of observations is rarely available in real-life applications. In the other tests examples, the QP-method still produced slightly better results than the MLE-method, but the difference was almost negligible. In the trialanine example, the leading eigenvalues of the generator where slightly better estimated by the MLE-method than by the QP-method, but this could possibly be changed by increasing the corresponding weights α_k, β_k, and γ_k in the functional (10).

The similar error behavior of the two methods could possibly be explained by the fact that both methods are affected by the sampling error, i.e., by the error inherent in the finite and noisy time-series data or the frequency matrix, respectively. We suppose that the sampling error is the limiting factor of both approaches.

From the view point of convergence, the QP-methods is superior because due to the quadratic objective function, it converges to an unique maximum, whereas the (enhanced) MLE-method may only converge to a local rather than global maximum. Furthermore, the convergence rate of the (enhanced) MLE-method is very slow, but we have observed that a sufficient accuracy was already attained after few iterations. However, each step in the iteration in both techniques involves very different operations (a full eigendecomposition in one case, a quadratic programming step in the other), hence the cost of each of these steps is very different as well.

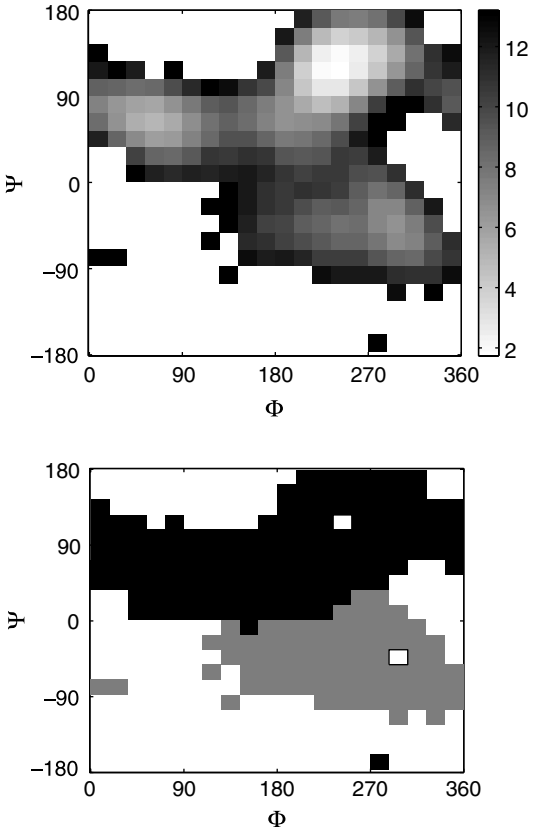

Fig. 10. Upper: box plot of the Gibbs-energy $-\log\pi_i$ where π is the stationary distribution computed via $\pi^T L_{\mathrm{MLE}} = 0$. Lower: decomposition of the torsion angles space into the two sets $\{i \in S: q_{AB}(i) \leqslant 0.5\}$ (depicted by the light grey boxes) and $\{i \in S: q_{AB}(i) > 0.5\}$ (the dark boxes), where q_{AB} is the committor function computed via (37). Results for an equidistant 20×20 box-discretization of the torsion angle space.

In the scope of this article, we have restricted ourselves to equidistant observation times. We remark that the QP-method in the presented form cannot simply be applied to time series with varying time lags, because in this situation the required transition matrix estimate is not available. The enhanced MLE-method, however, allows the efficient treatment of the case of non-equidistant observation times t_k by two slight modifications. Let $\{\tau_1, \ldots \tau_r\}, r > 1$ be the set of the different observation time lags with respect to the data Y and denote by $C(\tau_s), s = 1, \ldots, r$ the associated frequency matrices which provide the number of transitions between states with respect to the time lag τ_s, respectively. Then the conditional expectations in (17) can again be expressed as sums, e.g.,

$$\mathbb{E}_{L_0}[R_i(T)|Y] = \sum_{s=1}^{r} \sum_{k,l=1}^{d} c_{kl}(\tau_s) \mathbb{E}_{L_0}[R_i(\tau_s)|X(\tau_s) = l, X(0) = k].$$

Finally, the exact computation of the conditional expectations with respect to the different time lags requires the additional computation of the matrices $\Psi(\tau_1), \ldots, \Psi(\tau_r)$ (cf. (24)) which only has to be performed once in each EM-iteration step. That approach will be presented in a forthcoming paper.

Acknowledgements

We thank Eric Vanden-Eijnden for helpful discussions and the anonymous referees for useful comments, especially for pointing out additional references. One of the authors, E.D., is supported by the DFG priority program 1095 "Analysis, Modeling and Simulation of Multiscale Problems".

Appendix A

A.1. Two theorems on the existence of generators

The following Theorems are found in [16]. They give sufficient conditions for the existence of a generator of a given transition matrix.

Theorem 2. *Let P be a transition matrix and suppose that*

(a) $\det(P) \leqslant 0$, *or*
(b) $\det(P) > \prod_i p_{ii}$, or
(c) *there are states i and j such that j is accessible from i, but $p_{ij} = 0$.*

Then, there is no generator $L \in \mathfrak{G}$ such that $P = exp(L)$.

Theorem 3. *Let P be a transition matrix.*

(a) *If $\det(P) > \frac{1}{2}$, then P has at most one generator.*
(b) *If $\det(P) > \frac{1}{2}$ and $\|P - I\| < \frac{1}{2}$ (using any operator norm), then the only possible generator for P is the principal branch of the logarithm of P.*
(c) *If P has distinct eigenvalues and $det(P) > e^{-\pi}$, then the only possible generator for P is the principal branch of the logarithm of P.*

A.2. A simple example illustrating the effect of sampling errors

This example illustrates the influence of the sampling error on the optimal generator estimate; cf. the discussion in Section 5.1. The transition matrix of the generator

$$L = \begin{pmatrix} -0.2 & 0.2 \\ 0.2 & -0.2 \end{pmatrix}$$

with respect to the time lag $\tau = 1$ is

$$P(\tau) = \begin{pmatrix} 0.8352 & 0.1648 \\ 0.1648 & 0.8352 \end{pmatrix}.$$

Suppose that sampling according to the transition matrix produces the time series

time, t_n	0	1	2	3	4	5	6	7	8	9	10
state, $X(t_n)$	1	1	2	2	1	1	1	1	2	2	2

such that the corresponding frequency matrix is

$$C = \begin{pmatrix} 4 & 2 \\ 3 & 1 \end{pmatrix}.$$

According to this data, the transition matrix seems to be

$$\widehat{P} = \begin{pmatrix} 2/3 & 1/3 \\ 3/4 & 1/4 \end{pmatrix} \tag{A.1}$$

and since $\widehat{P} = \exp(\widehat{L})$ with

$$\widehat{L} \approx \begin{pmatrix} -0.5003 & 0.5003 \\ 0.3752 & -0.3752 \end{pmatrix} \in \mathfrak{G} \tag{A.2}$$

the best result we can expect to obtain based on the time series is $\widehat{L}$ instead of L. The errors $\|\widehat{P} - P\| \approx 0.2670$ and $\|\widehat{L} - L\| \approx 0.4916$ are caused by the time series and cannot be avoided by the two methods. However, these errors decrease if, according to the second test procedure, the frequency matrix is replaced by the virtual frequency matrix (33). Since in our example the stationary distribution is $\pi = (0.5,0.5)$, one obtains

$$C = \begin{pmatrix} 4 & 1 \\ 1 & 4 \end{pmatrix}.$$

The corresponding transition matrix

$$\widehat{P}_{virt} = \begin{pmatrix} 0.8 & 0.2 \\ 0.2 & 0.8 \end{pmatrix}$$

is obviously a better approximation of the true transition matrix P than (A.1), and the generator estimate

$$\widetilde{L} = \log(\widehat{P}_{\text{virt}}) \approx \begin{pmatrix} -0.2554 & 0.2554 \\ 0.2554 & -0.2554 \end{pmatrix}$$

is evidently better than (A.2). In fact, the new errors are only $\|P - \widehat{P}_{virt}\| \approx 0.0703$ and $\|L - \widetilde{L}\| \approx 0.1108$.

References

[1] M. Bladt, M. Sørensen, Statistical inference for discretely observed Markov jump processes, J.R. Statist. Soc. B 67 (2005) 395–410.

[2] D.T. Crommelin, E. Vanden-Eijnden, Fitting timeseries by continuous-time Markov chains: a quadratic programming approach, J. Comp. Phys. 217 (2006) 782–805.

[3] T. Müller, Modellierung von Proteinevolution, PhD thesis, Heidelberg, 2001.

[4] A. Hobolth, J.L. Jensen, Statistical inference in evolutionary models of DNA sequences via the EM algorithm, Statistical