HMC : When is it worth over RWMC ?

Course : Stochastic Simulation
Students : Aude Maier & Tara Fjellman
Fall 2024

1 Introduction

2 Core Theory

2.1 RWMC

A common shortfall of the simple random walk Metropolis proposal in the Metropolis-Hastings algorithm is the slow exploration rate of the state space. Much effort has been devoted in recent years to devise proposals with more efficient exploration rates (i.e "distant" proposals).

2.2 HMC

2.2.1 The Algorithm

HMC considers the Hamiltonian dynamics associated to a Hamiltonian function $H : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}, H = H(q,p)$, where the position variables $\{q_i\}$ are the random variables we want to sample and "fictitious" momentum variables $\{p_i\}$ are introduced.

The Hamiltonian dynamics are dictated by the equations

$$\frac{dq_i}{dt} = \frac{\partial H}{\partial p_i}, \tag{1}$$

$$\frac{dp_i}{dt} = -\frac{\partial H}{\partial q_i}, \tag{2}$$

for $i = 1, \ldots, d$. In general, the above equation can be understood as a conservation of the total energy of a system in time.

Samples of the q vector are then obtained by constructing a Markov Chain Monte Carlo algorithm with given invariant density $\pi(q)$ on the position variables $(q_1, \ldots, q_d)$. To do so, we introduce the potential energy $U(q) = -\log \pi(q)$, a kinetic energy $K(p) = \sum_{i=1}^d \frac{p_i^2}{2m_i}$, for some mass parameters $m_i, i = 1, \ldots, d$, and the Hamiltonian $H(q,p) = U(q) + K(p)$. Having introduced these functions, we can then simulate a Markov chain in which each iteration re-samples the momentum, evolves the Hamiltonian system for a certain time, and then performs a Metropolis-type acceptance-rejection step on the new position vector. More concretely, we consider the so-called Gibbs measure, given by

$$G(q,p) = \frac{1}{Z} e^{-H(p,q)},$$

where Z is the (unknown) normalizing constant. Notice that such a Gibbs measure naturally factorizes as:

$$G(q,p) = \frac{1}{\tilde{Z}} e{-U(q)} \frac{1}{\prod_{i=1}^d \sqrt{2\pi m_i}} e{-K(p)}$$

where $\frac{1}{Z} e^{-U(q)}$ is the probability density we are interested in sampling from, whereas the other factor is a multivariate Gaussian distribution $N(0, M)$ with $M = \operatorname{diag}(m_1, \ldots, m_d)$. Given the state q^n at iteration n, the idea of the algorithm is then to sample a momentum vector p^n from $N(0, M)$, and compute $H(q^n, p^n)$. The Hamiltonian system is then evolved starting from $q(0) = q^n, p(0) = p^n$, on a time interval $[0, T]$ using the dynamical equations for some arbitrary final time T, to obtain $(q(T), p(T))$. This state is then taken as the proposal state in a Metropolis-Hastings step to generate the new state q^{n+1}. For many problems of modern relevance, it is not possible to compute the dynamics exactly and numerical discretization is needed. A convenient time discretization scheme is the Verlet's method: the time interval $[0, T]$ is divided into N_t intervals of size $\varepsilon > 0$ and for each particle i the position q_i and momemtum p_i are updated as follows

$$p_i(t+\varepsilon/2) = p_i(t) - (\varepsilon/2)\frac{\partial U(q(t))}{\partial q_i}, \tag{3}$$

$$q_i(t+\varepsilon) = q_i(t) + \varepsilon\frac{p_i(t+\varepsilon/2)}{m_i}, \tag{4}$$

$$p_i(t+\varepsilon) = p_i(t+\varepsilon/2) - (\varepsilon/2)\frac{\partial U(q(t+\varepsilon))}{\partial q_i}. \tag{5}$$

2.2.2 Acceptance Rate

To explore how the acceptance rate behaves in HMC, we must consider the quantity $\exp\left[U\left(q^n\right)+K\left(p^n\right)-U\left(q^*\right)-K\left(p^*\right)\right]$ which appears in the expression for α in the Metropolis-Hastings acceptance probability. Using the definition $H(q,p) = U(q) + K(p)$ we can write this quantity as:

$$\exp\left(U\left(q^n\right)+K\left(p^n\right)-U\left(q^*\right)-K\left(p^*\right)\right) = \exp\left[H\left(q^n, p^n\right) - H\left(q^* p^*\right)\right]. \tag{6}$$

Since the Hamiltonian is conserved under the Hamiltonian dynamics :

$$\frac{dH}{dt} = \sum_i \frac{\partial H}{\partial p_i}\frac{dp_i}{dt} + \sum \frac{\partial H}{\partial q_i}\frac{\partial q_i}{dt} \tag{7}$$

$$= -\sum_i \frac{\partial H}{\partial p_i}\frac{\partial H}{\partial q_i} + \sum_i \frac{\partial H}{\partial q_i}\frac{\partial H_i}{\partial p_i} = 0, \tag{8}$$

we find by using this in Eq.6 that if integration is exact, the acceptance rate is always 1.

Under the assumption that the Hamiltonian dynamics is discretised, conservation is there in the best case on average. This implies the acceptance rate will be less than 1.

2.2.3 Convergence to Target Distribution

Gibbs measure invariance Under the assumption that there is no numerical error, we want to prove that the Gibbs measure is invariant for the chain generated by the hamiltonian dynamics.

This is equivalent to saying that the Gibbs measure π is the same before and after an evolution of t seconds from the hamiltonian dynamics. To prove this we first introduce hamiltonian dynamics operators φ, Φ acting respectively on the phase space and the Gibbs measure : $\varphi_t(q_s, p_s) = (q_{s+t}, p_{s+t}); \Phi_t[\pi_s] = \pi_{t+s} \quad \forall t \in \mathbb{R}$. The statement we want to prove can then be expressed as

$$\Phi_t[\pi_s](D) = \pi_{s+t}(D) \quad \forall D \in \mathcal{B}(\Omega), \forall s,t \in \mathbb{R}, \tag{9}$$

with Ω the phase space.

We can now write the left hand side of the equation as

$$\Phi_t[\pi_s](D) = \int_D \pi_{s+t}(q,p)\; dqdp \tag{10}$$

$$= \int_{\varphi_{-t}(D)} \pi_s(q,p)\; dqdp \tag{11}$$

$$= \pi_s(\varphi_{-t}(D)). \tag{12}$$

The final result is obtained using the fact that volumes in phase space are preserved by the hamiltonian dynamics (in conservative systems). This result is known as Liouville's theorem, but is mentioned as theorems 2.3 in [cite].

This implies specifically that $q_k \sim \pi$ for all $k \in \mathbb{N}$ if $q_0 \sim \pi$.

If the dynamics is discretised with the Velocity Verlet algorithm, the volume in phase space is preserved up to a small error, which is why the algorithm is used in practice [cite wikipedia].

3 Exploring a 2D example

3.1 Context

In this section we explore the performance of the presented algorithms on a 2D example. The target distribution is taken as $f_1(q_1, q_2) = e^{-\alpha(q_1^2+q_2^2-0.25)^2}$, with $\alpha > 0$ a parameter. This unormalised density is represented in figure Fig.1 for two different values of α. As it can be seen, the density has the shape of a doughnot and α controls its thickness. We expect the $\alpha = 1000$ case to be more difficult to sample from than the $\alpha = 10$ case, as the density is more localised.

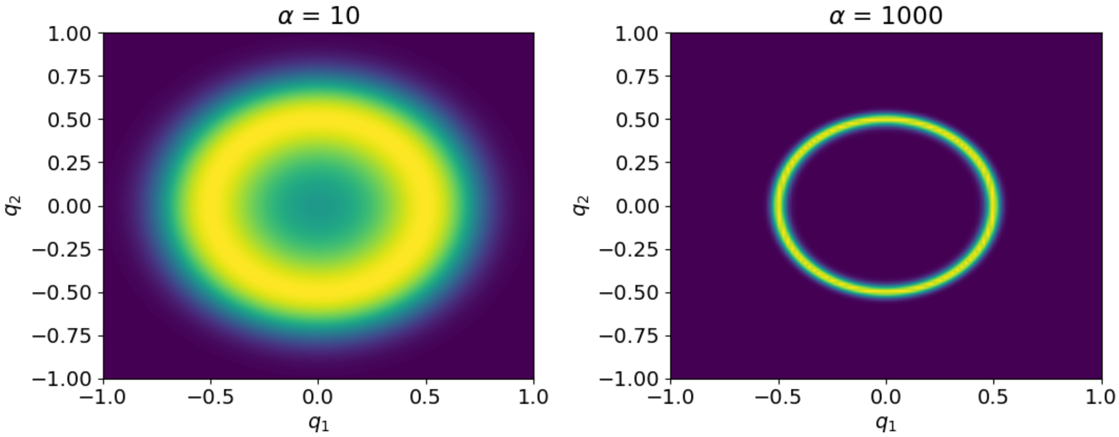

Fig 1: *Density considered in this section for two different values of α.*

3.2 RWMC Solution

As seen previously, the RWMC algorithms only depends on the step size. To find the best RWMC sampler we therefore explore the impact of the step size on performance.

Here and in the following, We decide to quantify performance throught the computation of a similarity based on the Jensen-Shannon divergence. This choice allows us to feed in a discretised version of f_1 (which we can normalise) and the empirical distribution of the samples generated by the algorithm, and get a similarity measure between the two.

The similarities associated to 3000 samples for the different step sizes are presented in figure Fig.2.

- both plots display a peak for a step size in the centre of the range considered (around 8.5×10^{-2} and 6.5×10^{-2} for $\alpha = 10$ and $\alpha = 1000$ respectively). This is expected as the step size is a tuning parameter that should be chosen to match the scale of the target distribution. - the peak is sharper (espescially on the right) for the $\alpha = 1000$ case, which is consistent with the fact that the density is more localised. - the value of the similarity is in all cases smaller than .5, which means that 3000 samples are too few to accurately estimate the target distribution. The value is higher in the $\alpha = 1000$ case, which can at first look surprising. Indeed, this case is meant to be harder than the $\alpha = 10$ one, but the fact that the density is more localised for $\alpha = 1000$ actually means that there are fewer places where the estimate and the target

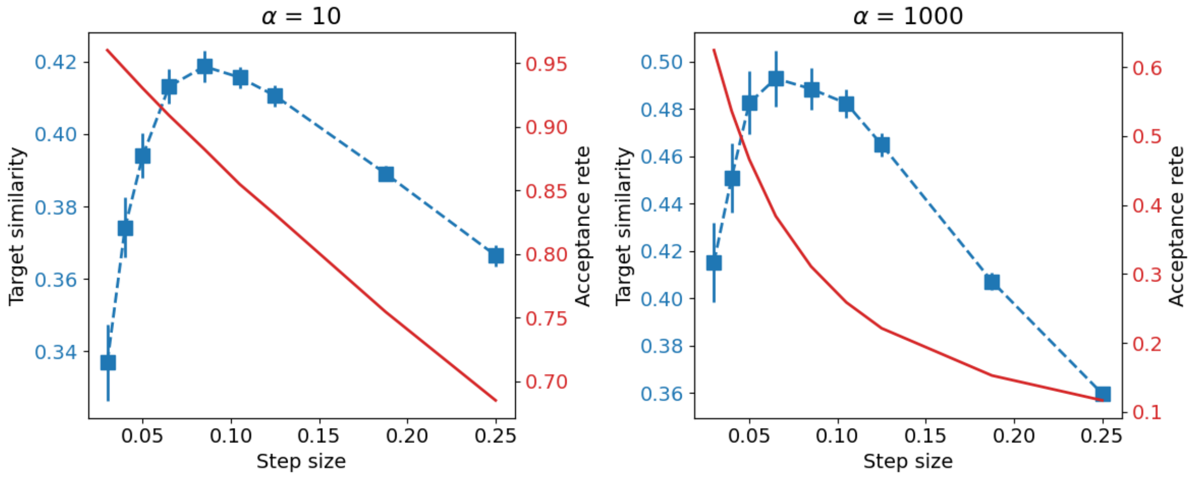

Fig 2: *Similarity as a function of RWMC step size for considered values of α.*

can differ, which can lead to a better similarity. It is therefore most important to consider the relative values of the similarities for the different step sizes, rather than the absolute values. - looking at acceptance rate, we see that they are of course monotonically deacreasing with step size. This means that the obtimal value corresponds with the best exploration-acceptance rate trade-off. The acceptance rate is higher and decreases slower for the $\alpha = 10$ case, which is consistent with the fact that the density is more spread out. The acceptance rate of the $\alpha = 1000$ case associated to the best step size is still around .55, which suggests that this case is still quite easy to sample from.

3.3 HMC Solution

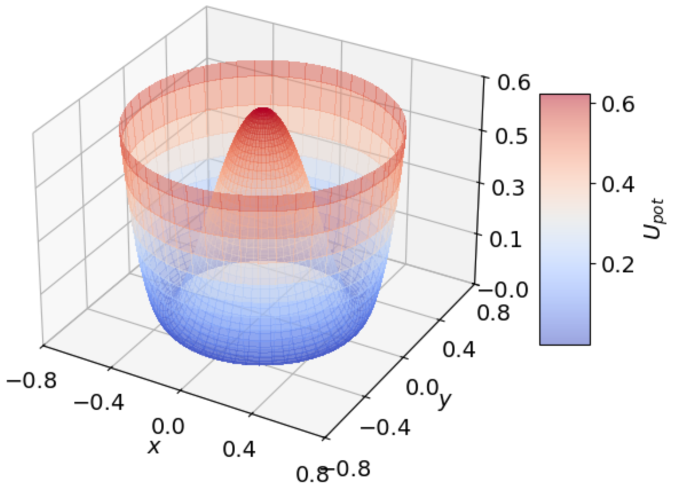

Fig 3: *Potential energy landscape associated to HMC algorithm for $\alpha = 10$. Version for $\alpha = 1000$ is identical, except scales are scaled by a factor of 100.*

Before exploring the impact of the different parameters of the HMC algorithm, we first present the potential energy landscape associated to the algorithm for this specific problem. The landscape is presented in figure Fig.3. The landscape has polar symmetry, meaning the trajectories will be some sort of symmetric oscillations. It has global minima at a radius of $\sqrt{0.25} = 0.5$ away from the centre and a local maxima at the centre. The only difference in the landscape for $\alpha = 1000$ w.r.t. the $\alpha = 10$ one is the scale of the potential. This means that the $\alpha = 1000$ will give rise to stronger potential forces, which translates the fact that the density is more localised.

3.3.1 Impact of Integration Time

The first parameter we explore is the integration time. For this parameter, we expect the optimal value to be the one that allows the sampler to explore the whole space, without being excessively long (as it would slow down sampling). The similarities associated to 3000 samples for the different integration times are presented in figure Fig.4. From a qualitative point of

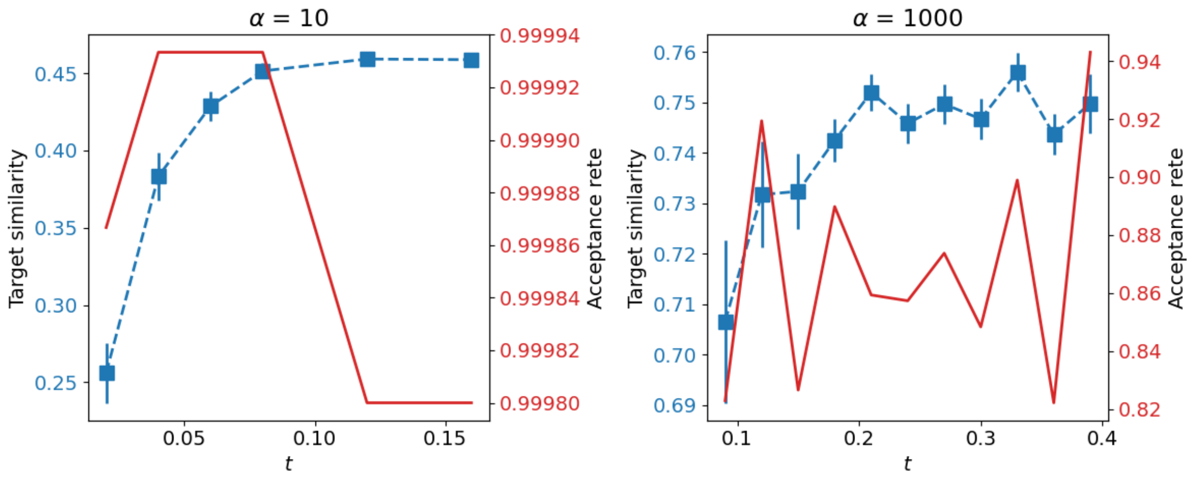

Fig 4: *Similarity as a function of HMC integration time for considered values of α.*

view, the behaviour is the same for both settings : similarity increases as a function of t untill a plateau is reached. Looking more closely, we notice that the curve is more noisy for the $\alpha = 1000$ case. This might be due to the smaller characteristic time of the dynamics, which makes the particle .... [FILL] - the plateau is reached quicker for the $\alpha = 10$ case, probably because the forces are stronger and the sampler can explore the space more quickly. For the $\alpha = 10$ case, $t = 8$ seems to strike a good balance for the mentionned trade-off. In the $t = 1000$ case ...[FILL]

3.4 Impact of Δt

The next parameter we explore is the time step. This parameter is important as it controls the accuracy of the integration. The goal is to find the largest time step thata allows for accurate integration, as this will speed up the sampler while allowing it to accept proposals frequently.

The similarities associated to 3000 samples for the different time steps are presented in figure Fig.5. The first characteristic feature the two plots share is the monotonous behaviour of

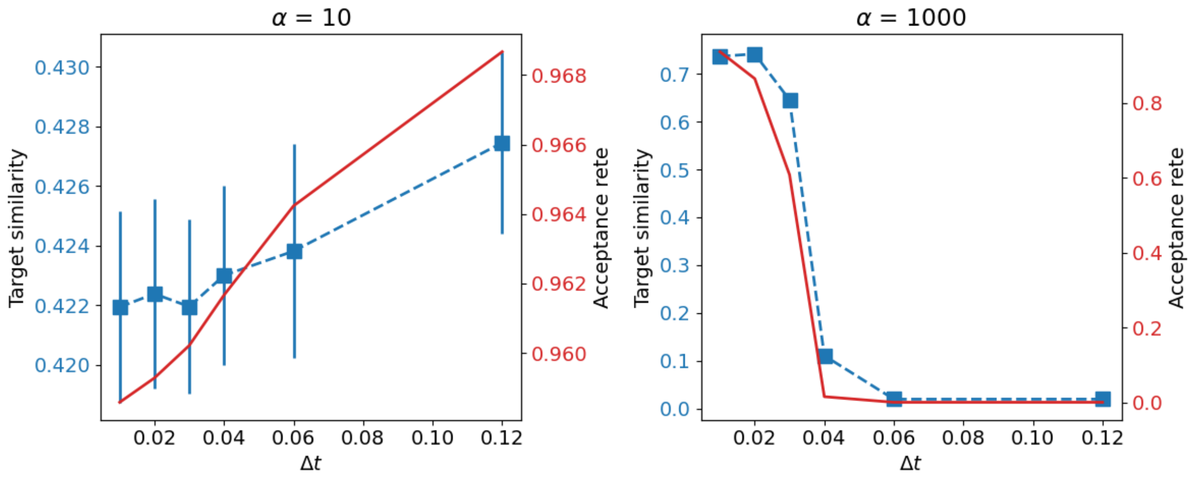

Fig 5: *Similarity as a function of HMC time step for considered values of α.*

target-similarity and acceptance rate. Indeed, as mentionned before, by making Δt grow, the simulation does not exactly leave the Hamiltonian invariant, and therefore makes the acceptance rate decrease. This in turn affects the similarity, as the obtained samples are less diverse and represent the target distribution less precisely. Looking more closely though, we notice the speed of decrease is really different : it is much quicker and of bigger scale for $\alpha = 1000$ (we basically reach 0 for the acceptance rate when $\Delta t = 6 \times 10^{-2}$). This is due to the characteristic time being shorter, and therefore requiring finer resolution. Indeed, one can only go up to $\Delta t = 3 \times 10^{-2}$ before really suffering in terms of performance, while for $\alpha = 10$ good performance is mantained in term of target similarity even when a single timestep of size t is taken (look at the scale of the vertical axis). This makes sense since the obtained algorithm is still some sort of upgraded version of RWMC where the general direction of the step is choosen smartly and RWMC already performed well over quite a broad domain of step sizes.

3.5 Impact of Mass Scale

We now turn to analysing the impact of the mass. We split this in terms of the mass scale and the mass symmetry. We do not consider the impact of off-diagonal terms as this would make the report too long.

What we expect to see is that the mass scale will control the speed of the dynamics (by affecting the particle's inertia). The similarities associated to 3000 samples for the different mass scales are presented in Fig.6. - In this case, both plots look qualitatively similar : both the acceptance rate and the similarity admit a global maxima at some value of the mass scale, and worsen on both sides of it. in both cases the acceptance rate reaches zero for a value of the order twice the optimal value. - bigger masses are better for alpha =1000 - reason for decrease on both end in terms of poor integration (citing GPT for stiff systems being harder) -

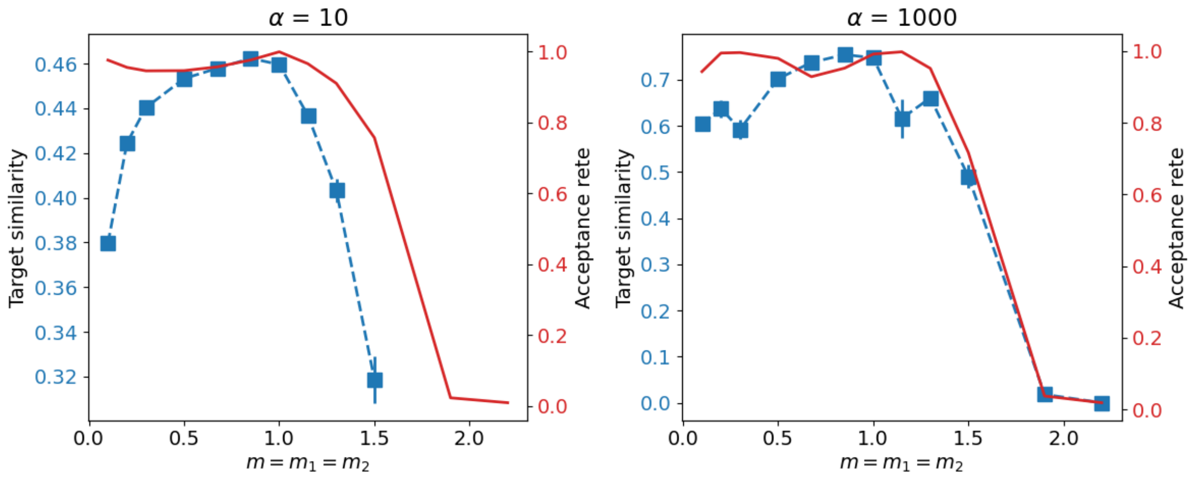

Fig 6: *Similarity as a function of HMC mass scale for considered values of α.*

3.6 Impact of Mass Symmetry

What we expect to see is that mass symmetry will control the isotropy/anisotropy of the trajectories. Indeed, if masses are different, the particle will have different inertia in different directions, which will affect the shape of the trajectories : in some direction the particle will react quickly to forces, while in others it will be more inert. We try to gain insight into this with help of Fig.7, in which the similarities associated to 3000 samples for both symmetric and asymmetric masses are represented. A first look at the plot shows that the similarity is better for the symmetric mass case. This makes sense given the target distribution (and therefore potential landscape) is isotropic. We would expect this to change if the distribution

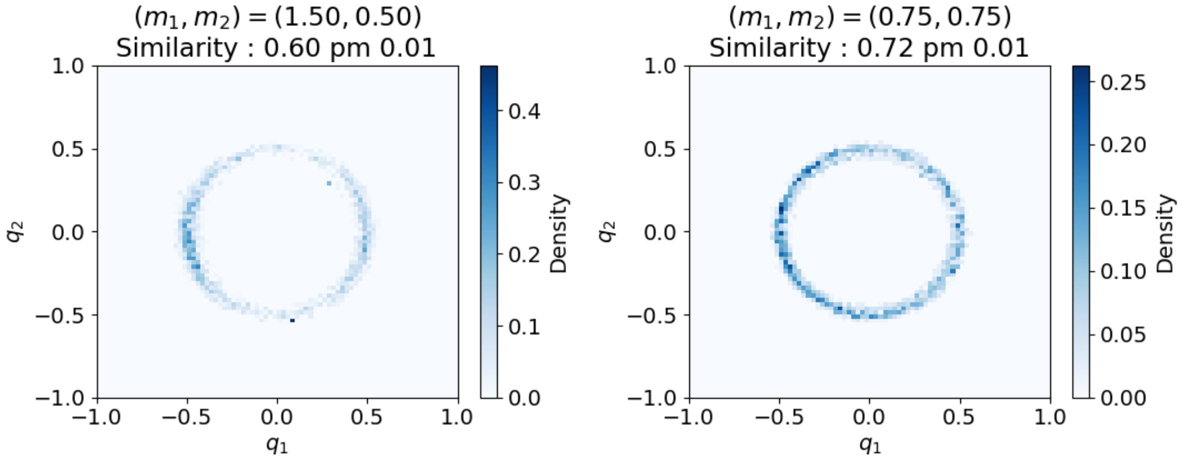

Fig 7: *Similarity for HMC samplers with asymmetric and symmetric masses for* $\alpha = 1000$.

was anisotropic. If the main axis of motion were not to be orthogonal, we would expect to find an advantage in properly choosing (non-zero) off-diagonal terms for the mass.

More precisely in our case, th plot associated to anisotropic masses reveils that left and right areas of the distributions are sampled more than the top and bottom ones. This seems to be a consequence of the fact that in that example $m_1 > m_2$, and therefore the particle's inertia is stronger along the horizontal axis than the vertical one.

3.7 Comment on interractions between parameters

In the previous sections, the analysis of HMC's parameters was limited to a one-factor-at-a-time study. This was motivated by the non-exhaustive character of the report and the fact that meaningfull insights were associated to these longitudinal analyses.

In reality however, each parameter does not act in a vacume. More specifically, qualitative explorations of simulations reveiled for example that : increasing mass slightly, required longer integration times for space-exploration to occur, and often allowed for a larger Δt to be used. The symmetric finding was found for lowering mass slightly. This has practical implications in finding global optimal choice of parameters. Indeed, when grid search for the parameters is untractable (i.e. in high dimensions), these interractions between the parameters make it hard to find globally optimal values. Our intuition is that in these cases one can proceed in the following way to find "good enough" parameters "easily" : .... - use assumptions on the potential to choose structure for the mass (isotropy, anysotropy etc.) - possibly do the same to derive a characteristic time for the oscillation - select the locally optimal mass scale after having set the time and Δt generously : the time larger than optimal (granting good mixing) and Δt smaller than optimal (granting precise integration) - with the new mass matrix, select the locally optimal integration time (shortest one preserving good enough mixing) - finally select the locally optimal Δt (the biggest for which the acceptance rate is good enough)

We do not expect this approximate tuning of the parameters to be a problem given that alternatives like RWMC breakdown in high dimensions. A locally optimal selection of parameters will probably still lead to a much more efficient sampler.

3.8 RWMC and HMC Comparisona

3.8.1 Target-Similarity as a Function of Sample-Size

To determine which algorithm is best to use for the studied 2D example, it is interesting to compare the results obtained with optimised versions of the RWMC and HMC algorithms as a function of sample size (matching the number of function evaluations). We do this in Fig.8.

Note that the optimal HMC parameters for the $\alpha = 10$ case were not found following the algorithm described above. Indeed in this special case, the problem is so simple that taking a single timestep of size t is enough to get a good similarity. The improvement obtained by taking more timesteps is therefore irrelevant when compared to the computation burden. In general, the same qualitative behaviour is observed for all algorithms : the similarity quickly increases at the start but then sees deminishing returns characteristic of concave functions. In both cases RWMC outperforms HMC in terms of similarity. The difference is largest in the $\alpha = 10$ case, again, due to how simple the target function is. For $\alpha = 1000$, the computational overhead (of a factor 7) is for a big part compensated. We can easily imagine this trend continuing with the dimension of the distribution, and therefore HMC vastly outperform RWMC in these settings.

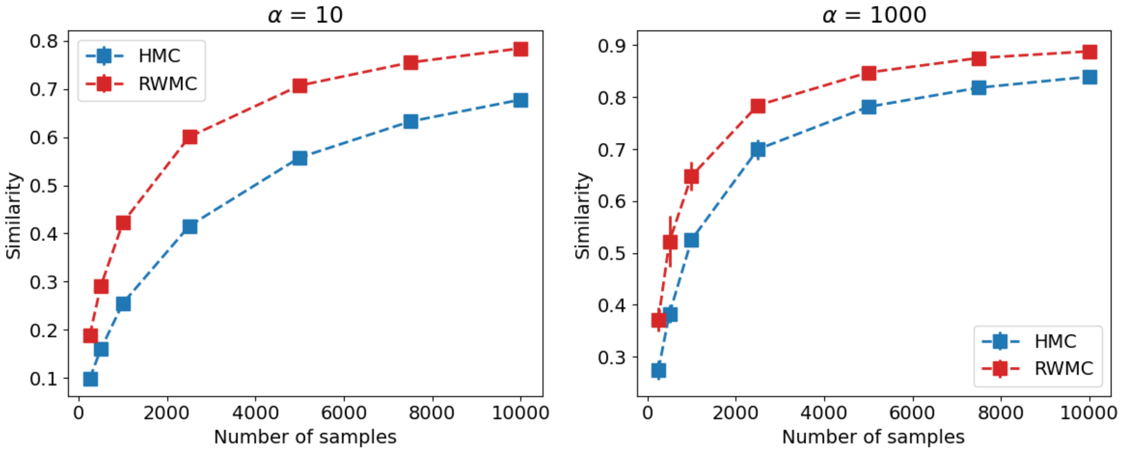

Fig 8: *Similarity as a function of sample-size for RWMC and HMC samplers and considered values of α. The sample-size represented on the x axis is that associated to the HMC sampler. The associated RWMC sample-size is obtained by matching the number of function evalutations of the HMC sampler.*

3.8.2 Effective Sample-Size

4 US Birthweight Data

4.1 HMC Solution

4.2 RWMC Solution

5 Conclusion

Aknowledgements

References

A Commented Code Snippet

[cite stack exchange for format]

```
    if transactions: Transaction.create_transactions() # if transactions =
                                                "true"
    node.generate_emptyState() # empty state for all nodes
    S.initial_events() # initiate initial events to start with

    while not queue.isEmpty() and clock <= targetTime:
      next_e = queue.get_next_event()
```

```
 clock = next_e.time # move clock to the time of the event
 Event.execute_event(next_e)
 Queue.remove_event(next_e)

print results
```

B Rejection Sampling Attempt

As an alternative to HMC we consider rejection sampling. We therefore want to find a function $g(q)$ and a constant C such that the following inequality holds for all q:

$$\tilde{f}(q) = e^{q^T X^T (y-1_n)} e^{-1_n^T \log[1+\exp(-x_i^T q)]_{n\times 1}} e^{-\frac{1}{2}q^T \Sigma^{-1} q} \leq Cg(q), \tag{13}$$

where we have denoted $\Sigma = \text{Diag}(\sigma_1^2, ..., \sigma_p^2)$. Given that

$$\log[1+\exp(-x_i^T q)] \leq \log(2) - x_i^T q, \tag{14}$$

$$e^{-\sum_i \log[1+\exp(-x_i^T q)]} = \prod_i \frac{1}{1+\exp(-x_i^T q)} < 1, \tag{15}$$

we can simplify the problem to finding a function $g(q)$ such that

$$\tilde{f}(q) \leq 2^{-n} e^{-q^T X^T 1_n} e^{q^T X^T (y-1_n)} e^{-\frac{1}{2}q^T \Sigma^{-1} q} = 2^{-n} e^{q^T b} e^{-\frac{1}{2} q^T \Sigma^{-1} q} =: Cg(q), \tag{16}$$

with $b = X^T(y - 2_n)$.

By completing the square in the exponent of $Cg(q)$, we can write it in terms of a Multivariate Gaussian distribution with mean $\mu = \Sigma b$ and covariance Σ. Indeed :

$$e^{-\frac{1}{2}(q-\mu)^T \Sigma^{-1}(q-\mu)} = e^{-\frac{1}{2}\mu^T \Sigma^{-1}\mu} e^{q^T \Sigma^{-1}\mu} e^{-\frac{1}{2}q^T \Sigma^{-1} q} \tag{17}$$

$$\implies \tilde{f}(q) \leq 2^{-n} e^{\frac{1}{2}\mu^T \Sigma^{-1}\mu} e^{-\frac{1}{2}(q-\mu)^T \Sigma^{-1}(q-\mu)}. \tag{18}$$

Using now the normalisation constant of the Multivariate Gaussian distribution

$$\sqrt{(2\pi)^p |\Sigma|} = \int_{\mathbb{R}^p} e^{-\frac{1}{2}(q-\mu)^T \Sigma^{-1}(q-\mu)} \, dq, \tag{19}$$

we can define g and C as

$$g(q) = \frac{1}{\sqrt{(2\pi)^p |\Sigma|}} e^{-\frac{1}{2}(q-\mu)^T \Sigma^{-1}(q-\mu)}, \tag{20}$$

$$C = 2^{-n} e^{\frac{1}{2}\mu^T \Sigma^{-1}\mu} \sqrt{(2\pi)^p |\Sigma|} = 2^{-n} \sqrt{(2\pi)^p |\Sigma| e^{\mu^T \Sigma^{-1} \mu}}. \tag{21}$$