Speeding up the Zig-Zag process

Giorgos Vasdekis[1] and Gareth O. Roberts[2]

[1] Department of Statistical Science, London, University College London, U.K.
G.Vasdekis@ucl.ac.uk,
[2] Department of Statistics, University of Warwick, U.K.

Abstract. The Zig-Zag process is a Piecewise Deterministic Markov Process (PDMP), efficiently used for simulation in an MCMC setting. A generalisation of this process, the Speed Up Zig-Zag (SUZZ) process, was later suggested in Vasdekis G. and Roberts G. O. (2023+) [28] as a way to explore the tails of the distribution faster, making it an ideal candidate for heavy tailed targets. In this article we will describe the SUZZ process, we will review the main theoretical results and we will present a numerical study on some more practical models than the ones discussed in Vasdekis G. and Roberts G. O. (2023+) [28], showing that the advantages of using SUZZ may also extend to lighter tailed targets.

Keywords: Piecewise Deterministic Markov Process, Markov Chain Monte Carlo

1 Introduction

Markov Chain Monte Carlo (MCMC) is an important technique, widely applied in Bayesian statistics when one tries to numerically estimate intractable integrals with respect to a posterior distribution. MCMC offers a solution to this problem by constructing a Markov chain that has the posterior as invariant and then using samples from the chain as samples distributed approximately according to the law of the posterior. One then uses these samples to estimate the intractable integral. Traditional MCMC algorithms, such as MALA or Random Walk Metropolis (RWM) are constructed to be time-reversible. On the other hand, there is evidence that non-reversible Markov chains can sometimes outperform reversible ones (see for example [20, 17, 24, 13, 21, 3, 18]). Piecewise deterministic Markov processes (PDMPs) have recently been used as a way to construct non-reversible MCMC algorithms. These are processes that move deterministically for a random period of time before randomly jumping to a different type of deterministic movement. This partially deterministic behaviour gives them a notion of momentum which can accelerate the state space exploration.

General literature on PDMPs and their applications to MCMC includes [19, 26, 5, 4, 9, 11, 8, 7, 15, 16, 25] etc. The two PDMPs that first appeared in the literature of MCMC were the Bouncy Particle Sampler [12] and the Zig-Zag sampler [6]. The latter was suggested in [6] as an algorithm particularly well suited in a Bayesian setting involving large data sets. In [10] the authors prove ergodicity and exponential ergodicity of the Zig-Zag process in arbitrary dimension. As in many other MCMC algorithms, the exponential ergodicity result must assume that the target distribution has tails lighter than some exponential distribution. In [28], the authors proved that the Zig-Zag process fails to be exponentially ergodic when the target distribution has tails heavier than any exponential distribution. In fact, polynomial rates of L^2-convergence have been proven in [1] for the process in arbitrary dimension, while [27] proves tight polynomial rates of convergence in the total variation distance, for the one-dimensional process, when the target has tails that decay like a Student distribution.

In order to overcome the problem of Zig-Zag's slow mixing on heavy tails, [28] introduced a variant of the Zig-Zag process, called Speed Up Zig-Zag (SUZZ). Instead of only permitting the process to move with unit speed, the SUZZ process has a positive, position-dependant speed, which assists exploration of the tails and subsequent return to the high density areas of the distribution more rapidly. Indeed, exponential ergodicity results were established in [28] for the SUZZ process, and, when applied to some heavy tailed toy models, the process was shown to provide significant computational advantages, even against state of the art algorithms. In this article, we will provide numerical results that show that the SUZZ process can be efficient even on lighter tails.

The rest of the article is organised as follows. In Section 2 we describe the SUZZ process. In Section 3 we recall the theoretical results proved in [28] concerning the process. Finally, in Section 4 we provide a numerical comparison between the SUZZ process, the original Zig-Zag and a Metropolis-Hastings algorithm in the context of Bayesian logistic regression.

2 The SUZZ process

In this section we will describe the SUZZ process, introduced in [28]. This is a d-dimensional, continuous time process that moves in straight lines, parallel to vectors of the form $\{-1,+1\}^d$, but the speed in which the process traverses these straight lines depends on its current position. Typically the speed $s \in C^1\left(\mathbb{R}^d\right)$ of the process will increase the more the process moves away from the mode of the posterior, allowing for a faster exploration of the tails. The state of the process consists of a position $x \in \mathbb{R}^d$ and a direction $v = (v_1, ..., v_d) \in \{-1,+1\}^d$. For a random period of time, the direction v remains constant, while the x-component of the process deterministically follows the straight line parallel to v. This random period of time is given as the first arrival time of a non-homogeneous Poisson process. Afterwards, the direction v is updated and the process starts following a different straight line, etc.

More precisely, let us assume that one wants to target a d-dimensional posterior with density

$$\pi(x) = \frac{1}{Z}\exp\{-U(x)\}, \tag{1}$$

with $U \in C^1$ and $Z = \int_{\mathbb{R}^d} \exp\{-U(y)\}dy < \infty$. Let us assume that the process has **speed function** $s \in C^1$. For all $i \in \{1, ..., d\}$, let's consider the functions

$$\lambda_i(x, v) = \max\{0, v_i A_i(x)\} + \gamma_i(x), \tag{2}$$

where

$$A_i(x) = s(x)\partial_i U(x) - \partial_i s(x), \tag{3}$$

and ∂_i denotes the partial derivative with respect to the i-coordinate. Here γ_i can be picked by the user of the algorithm and can be any non-negative, locally bounded, integrable function that only depends on x. A natural choice is $\gamma_i(x) = 0$ for all $x \in \mathbb{R}^d$.

The state space of the process will be $E = \mathbb{R}^d \times \{-1, 1\}^d$. When the process is at point $(x, v) \in E$, with $x \in \mathbb{R}^d$ and $v \in \{-1, 1\}^d$, the x-component will move along the straight line $\{x + vt, t \geq 0\}$ with speed function s that depends on the current position. Formally, the x-component of the process will follow a deterministic path X_t which solves the system of Ordinary Differential Equations (ODE)

$$\begin{cases} \dfrac{d}{dt} X_t = v \cdot s(X_t), t \geq 0 \\ X_0 = x. \end{cases} \tag{4}$$

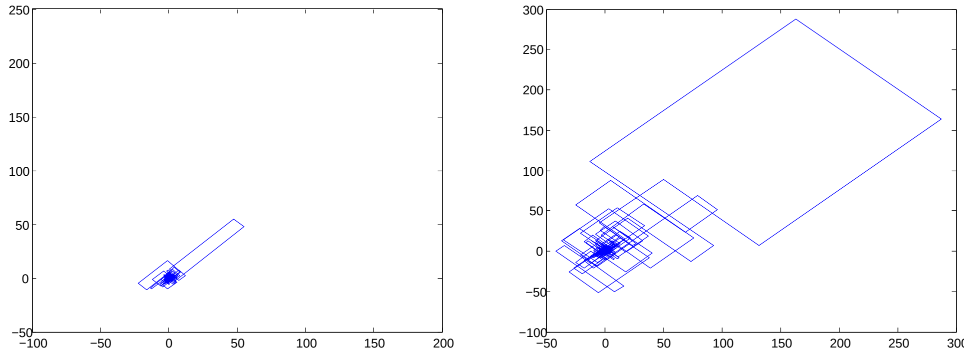
For each coordinate $i \in \{1, ..., d\}$, we let T_1^i denote the first event of a non-homogeneous Poisson Process with rate $m_i(t) = \lambda_i(X_t, v)$.
Let $T_1 = \min_{i\in\{1,...,d\}} T_1^i$ and $j = \arg\min_{i\in\{1,...,d\}}\{T_1^i\}$. The SUZZ process is defined until time T_1 to be $(X_t, V_t)_{t<T_1}$, where X_t is the solution of (4) until time T_1 and $V_t = v$. At time T_1 the direction V_{T_1} of the process switches from v to $F_j(v) = (v_1, ..., v_{j-1}, -v_j, v_{j+1}, ..., v_d)$. Then the process starts again from the new starting point $(X_{T_1}, F_j(v))$ and the x-component evolves as the solution of the ODE (4), with starting point X_{T_1} and direction $F_j(v)$. The x-component follows this ODE until the random time T_2, defined as the first arrival time of a Poisson processes, in a similar fashion to T_1. Then the direction of the process is again updated and the process starts again, etc.

The algorithmic description of a d-dimensional SUZZ process targeting a d-dimensional posterior distribution with density is given by the following algorithm.

Algorithm 1 (Speed Up Zig-Zag) *1. Set* $t = 0$
2. Start from point $(X_t, V_t) = (x, v) \in \mathbb{R}^d \times \{-1, +1\}^d$.
3. The process (X_{t+u}, V_{t+u}) *moves according to the deterministic ODE system*

$$\begin{cases} \dfrac{d}{du} X_{t+u} = v \cdot s(X_{t+u}), u \geq 0 \\ X_t = x, \end{cases} \tag{5}$$

and $V_{t+u} = v$, $u \geq 0$. *Let* $\{\Phi_u(x, v), u \geq 0\}$ *be the solution over time* u *of the ODE (5) with starting point* $x \in \mathbb{R}^d$ *and* $v \in \{-1, +1\}^d$.



(a) Zig-Zag process

(b) Speed Up Zig-Zag process

Fig. 1: Trace plots of ZZ and SUZZ process with speed $s(x) = \left(1 + \|x\|_2^2\right)^{1/2}$, targeting a two-dimensional Cauchy with positive correlations.

4. *For every coordinate $i \in \{1, ..., d\}$, define the function $\lambda_i : \mathbb{R}^d \times \{-1, +1\}^d$ as in (2). For all i, consider the non-homogeneous Poisson process with intensity $\{m_i(u) = \lambda_i(\Phi_u(x, v), v), u \geq 0\}$.*
5. *Let τ_i be the first arrival time of the i'th Poisson process, i.e. for all $t_0 \geq 0$, $\mathbb{P}(\tau_i \geq t_0) = \exp\{-\int_0^{t_0} m_i(u)du\}$. Let $j = argmin\{\tau_i, i = 1, ..., d\}$ and $\tau = \tau_j$ the first arrival time of all the processes.*
6. *For $u \in [0, \tau)$ set $X_{t+u} = \Phi_u(x, v)$ and $V_{t+u} = v$.*
7. *Set $t = t + \tau$, $x = \Phi_\tau(x, v)$ and $X_t = x$.*
8. *If $v = (v_1, ..., v_d)$, set a new $v = (v_1, ..., v_{j-1}, -v_j, v_{j+1}, ..., v_d)$ and set $V_t = v$.*
9. *Repeat from the Step 2.*

Remark 1. It is important to note that for one to use the SUZZ algorithm to target π, one needs to use the entire path $(X_t)_{t\geq 0}$ and not only the switching points X_{T_k} since these will be biased towards the tails of the distribution. In practice one can specify a $\delta > 0$ and use the points $X_{m\cdot\delta}, m = 1, 2, 3, ...$ as the output of the MCMC algorithm.

When one tries to implement the SUZZ process in a computer, two issues arise. The first is efficiently simulating the switching times of the process from the Poisson process, a common problem in the PDMP literature. For recent work on how one can practically tackle this problem we refer the reader to [15, 16, 25]. The second issue is that one needs to be able to simulate the deterministic dynamics of the process directly, meaning that one should use a speed function s such that the ODE (4) admits a closed form solution. A family of speed functions that lead to closed form solutions is

$$s(x) = \left(1 + \|x\|_2^2\right)^{\frac{1+k}{2}}, \tag{6}$$

for any $k = 0, 1, 2, 3, ...$ For more details on how one can explicitly solve the ODE for this class of speed functions, the reader is referred to [28].

It is interesting to note that for s as in (6) with $k > 0$, and more generally when the speed function s grows super-linearly, the solution to the ODE (4) explodes and reaches infinity in finite time. This could be of great use since it can allow the algorithm to quickly reach and explore the tails of the target. At the same time, after the process reaches the tails of the target, it can quickly come back to the mode, leading to a very stable algorithmic behaviour. Part of the theory of [28] was to prove that using this type of ODE is mathematically feasible to implement. In order to do that, one needs to prove that even though the deterministic dynamics may explode in finite time, the rate of switching direction is sufficiently large and the Poisson process will a.s. force the SUZZ process to switch direction before reaching infinity. This way, the process is a.s. non-explosive. To the best of our knowledge this is the first time explosive dynamics are being used in PDMP Monte Carlo.

3 Theoretical Results

We now recall the main theoretical results of [28], which prove that under conditions on s and U, the SUZZ process exhibits exponentially fast convergence to the target distribution in total variation distance. Indeed one can prove that for practical choices of speed functions, the convergence is exponential even for some heavy tailed targets, such as targets that decay like $\exp\{-\|x\|^a\}$ for some $a < 1$ (see [28] for more details).

In order to guarantee the theoretical properties of SUZZ process, one must make the following assumptions.

Assumption 1 (Speed Growth) $\lim_{\|x\|\to\infty} \|x\|^{d-1} s(x) \exp\{-U(x)\} = 0.$

Assumption 2 (Rates Growth) *Assume that for the refresh rates there exists* $\bar{\gamma}$ *such that for all* $i \in \{1, ..., d\}, x \in \mathbb{R}^d$, $\gamma_i(x) \leq \bar{\gamma}$.
Furthermore, assume that there exists $R > 0$ *and* $A > 0$ *so that for all* $\|x\| \geq R$,

$$\sum_{i=1}^{d} |A_i(x)| > A > \max\{3d\bar{\gamma}, 4d(d-1)\bar{\gamma}\}. \tag{7}$$

Furthermore, the authors in [28] make the following assumption.

Assumption 3 *If we iteratively define the functions* $h_n : [0, +\infty) \to [0, +\infty)$ *such that*

$$h_0(x) = x \tag{8}$$

and for $n \geq 1$

$$h_n(x) = \log(1 + h_{n-1}(x)). \tag{9}$$

then there exists an $n \in \mathbb{N}$ *such that*

$$\lim_{\|x\|\to\infty} \frac{h_n\left(s(x)\|\nabla\left(U(x) - \log s(x)\right)\|_1\right)}{U(x) - \log s(x)} = 0. \tag{10}$$

Furthermore, assume that for all $j \in \{1, ..., d\}$, $A_j \in C^1$ and ,

$$\lim_{\|x\|\to\infty} \frac{s(x)}{\sum_{k=1}^{d} |A_k(x)|} \sum_{i=1}^{d} \sum_{j=1}^{d} \frac{|\partial_i A_j(x)|}{1 + |A_j(x)|} = 0. \tag{11}$$

In what follows we will be writing μ to denote the measure on E such that

$$\mu = \pi \otimes \frac{1}{2^d} \sum_{v \in \{-1,+1\}^d} \delta_v, \tag{12}$$

i.e. the product measure between π and the uniform distribution on $\{-1, +1\}^d$, which has π as marginal in the x-component and the uniform as marginal on the v-component. Then, [28] proves the following.

Theorem 1. *Let $(Z_t)_{t\geq 0} = (X_t, \Theta_t)_{t\geq 0}$ be a SUZZ process with speed function $s \in C^2$ bounded away from 0.*

- *Assume that the rates satisfy (2) and Assumptions 1, 2 and 3 hold. Then the SUZZ process is a.s. non-explosive and has the measure μ in (12) as invariant.*
- *Assume further that the function $U - \log s \in C^3$ and has a non-degenerate local minimum, i.e. there exists a local minimum x_0 such that the Hessian of $U - \log s$ is strictly positive definite at x_0. Finally, assume that μ is a probability measure. Then the SUZZ process is exponentially ergodic, i.e. there exists $\rho < 1$ and $M : E \to [1, +\infty)$ such that for all $(x, v) \in E$,*

$$\|\mathbb{P}_{x,v}\left((X_t, V_t) \in \cdot\right) - \mu(\cdot)\|_{TV} \leq M(x, v)\rho^t.$$

- *Assuming the assumptions of the previous bullet, let $\{Y_n, n \geq 0\}$ be any skeleton of the SUZZ process (i.e. for some $\delta > 0$, $Y_n = Z_{n\cdot\delta}$ for all $n \in \mathbb{N}$) and let $f : E \to \mathbb{R}$ such that there exists an $\epsilon > 0$ with $\mathbb{E}_\mu[f^{2+\epsilon}] < \infty$. Then, a CLT result holds, i.e. there exists a $\gamma_f^2 \in [0, \infty)$ and $\tilde{Z} \sim \mathcal{N}(0, \gamma_f^2)$ such that*

$$\sqrt{n}\left(\frac{1}{n}\sum_{k=1}^{n} f(Y_k) - \mu(f)\right) \xrightarrow[D]{n\to\infty} \tilde{Z}.$$

The second convergence result proved in [28] is of similar flavour and makes assumptions that are easier to verify in practice. On the other hand these assumptions essentially force the target to have lighter tails (i.e. one must have that $\liminf_{\|x\|\to\infty} \|\nabla U(x)\| > 0$). However, light tailed targets can be very commonly found in applications of Bayesian statistics.

Assumption 4 *Assume that $U - \log s \in C^2$ and there exists an $\tilde{M} > 0$ such that the rates γ_i as in (2) satisfy $\gamma_i(x) \leq \tilde{M}s(x)$ for all $x \in \mathbb{R}^d$. Assume further that for some $n \in \mathbb{N}$, if h_n as in (9) and Hess denotes the Hessian matrix, then*

$$\lim_{\|x\|\to\infty} \frac{h_n\left(\|\nabla(U(x) - \log s(x))\|\right)}{U(x) - \log s(x)} = 0, \quad \lim_{\|x\|\to\infty} \frac{\|Hess(U(x) - \log s(x))\|}{\|\nabla\left((U(x) - \log s(x)\right)\|} = 0,$$

and that there exists $R > 0$ and $A > 0$ so that for all $\|x\| \geq R$,

$$\|\nabla(U(x) - \log s(x))\|_1 > A > \max\{3d\tilde{M}, 4d(d-1)\tilde{M}\}. \tag{13}$$

Theorem 2. *Let $(Z_t)_{t\geq 0} = (X_t, \Theta_t)_{t\geq 0}$ be a SUZZ process with speed function $s \in C^2$ bounded away from 0.*

- *Assume that the rates satisfy (2) and Assumptions 1 and 4 hold. Then the SUZZ process is non-explosive and has the measure μ in (12) as invariant.*
- *Assume further that the function $U - \log s \in C^3$ and has a non-degenerate local minimum, in the sense of Theorem 1. Finally, assume that μ is a probability measure. Then the SUZZ process is exponentially ergodic.*
- *Assuming the assumptions of the previous bullet, let $\{Y_n, n \geq 0\}$ be any skeleton of the SUZZ process and let $f : E \to \mathbb{R}$ such that there exists an $\epsilon > 0$ with $\mathbb{E}_\mu[f^{2+\epsilon}] < \infty$. Then, the CLT result of Theorem 1 holds.*

One should note that the assumptions of Theorem 2 are essentially the same as the ones made in [10] to prove exponential ergodicity for the original Zig-Zag process. This seems to suggest that the SUZZ process with any reasonable speed function should at least not perform much worse than the original Zig-Zag on any target where the latter one performs well. This allows us to see the speed function as a tuning parameter for the process, which, if chosen carefully, could lead to significant computational advantages. In [28] the authors show through some toy model simulations that these advantages can occur in heavy tailed targets. In the next section we will show that these advantages can occur even on light tailed targets, more commonly encountered in Bayesian applications (see for example [14]).

4 Numerical Examples

Below we present a numerical example where we target the posterior occurring from a logistic regression model. This is commonly used as a benchmark problem for MCMC algorithms (see for example [14]). In this model, conditional on a d-dimensional parameter $\beta = (\beta_1, ..., \beta_d)$ and given d-dimensional covariates $x^j = (x^j_1, ..., x^j_d) \in \mathbb{R}^d$, where $j = 1, ..., n$, the binary variable $y^j \in \{0, 1\}$ has distribution given by

$$\mathbb{P}\left(y^j = 1\right) = \frac{1}{1 + \exp\left\{-\sum_{i=1}^d \beta_i x^j_i\right\}}.$$

We considered this model for dimension $d = 2$ and $d = 16$. For both cases we assigned β a prior distribution such that for all i, β_i are i.i.d. $\mathcal{N}(0, 100)$. This is a fairly non-informative prior. We generated data using the values of $\beta_1 = -\log(4)$ and $\beta_2 = 0.5$ for both $d = 2$ and $d = 16$ cases. We also set $x^j_1 = 1$ for all j and we generated x^j_2 according to $\mathcal{N}(0, 5)$. For the $d = 16$ case, we also set $\beta_3 = 10^{-3}$, and $\beta_i = 1$ for $i = 4, ..., 16$, while $x^j_i \sim \mathcal{N}(0, 5)$ for all

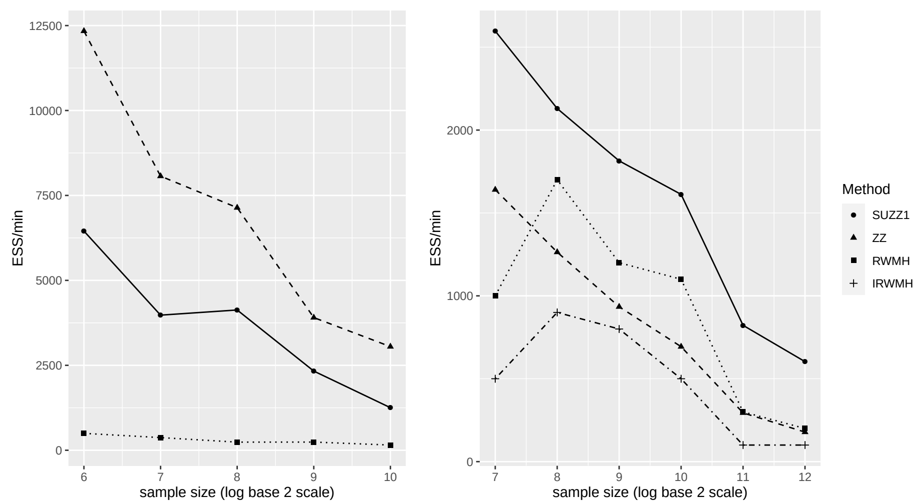
$i = 3, ..., 16$. We considered the posterior distributions on β for various number of observations. More specifically, we considered $n = 2^6, 2^7, ..., 2^{10}$ for the $d = 2$ case and $n = 2^7, ..., 2^{12}$ for the $d = 16$ one.

We compared the performance of the original Zig-Zag process (ZZ) with the one of the SUZZ(1) process, where SUZZ(k) denotes the SUZZ process with speed function given by (6). Note that SUZZ(1) has explosive deterministic dynamics, but the actual process a.s. does not explode. A general way to construct the deterministic dynamics for this type of speed functions can be found in [28]. We also compared these algorithms against a Random Walk Metropolis (RWM) algorithm with Normal distribution as proposal. For the covariance matrix of the random walk's proposal we used an approximation of the asymptotic covariance matrix of the maximum likelihood estimators of β's (see for example [22]). For the $d = 16$ case, we also compared against a RWM with diagonal covariance matrix, which we will call Independent RWM (IRWM). Following the guidelines of [23], all the proposals for the RWM and IRWM algorithms were tuned so that the acceptance ratio was close to 0.234. For each posterior and each of the algorithms presented, we simulated ten independent realisations of the process, until $N = 10^5$ switches of direction occurred (or for $N = 10^5$ steps in the case of Random Walk). As a sample from the PDMPs we used the position of the process every δ time units, as mentioned in Remark 1. Here δ was chosen after an initial run of the algorithm such that the sample size was roughly equal to the number of direction switches ($N = 10^5$). We refer the reader to [28] for more details on why choosing δ this way. All simulations were performed using `MATLAB` in a computer with i7-8550U CPU and 1.80 GHz.

We present our results in Figures 2 and 3. The x-axis of the Figures is the number of observed data used to construct the posterior. For every such posterior and for each algorithm, we present the average ESS per minute of implementation time (Figure 2) or the average ESS per likelihood evaluation (Figure 3). We should note here that for presentational convenience, the ESS in Figure 3 are stated in scale 10^{-3}, i.e. one should multiply with 10^{-3} to get the actual value. The averages are taken over the ten independent implementations of the respective algorithm. For all algorithms, we compute the ESS using the routine `mcmcse` of `R`. Both figures report results both for dimension $d = 2$ and $d = 16$.

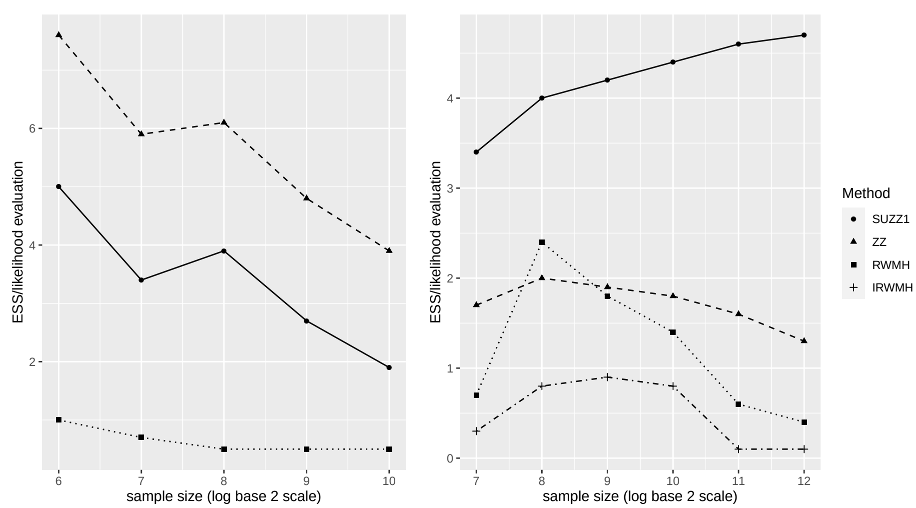
In dimension $d = 2$, all three algorithms provided a small Mean Square Error (MSE), which can increase our trust that all algorithms converged to the right distribution. In terms of performance comparison, we observe that both PDMPs vastly outperformed the RWM, while the ZZ algorithm outperformed SUZZ(1).

The picture is different when the dimension of the parameter space increased. In particular, the IRWM consistently underperformed compared to the other algorithms, the RWM and the ZZ performed relatively similar, while SUZZ(1) outperformed all three other algorithms. Indeed SUZZ(1) seems to consistently outperform ZZ in this higher dimensional case, irrespective of the number of observations. This shows that using a speed function in the context of PDMP algorithms can lead to significant benefits, even when the target has light tails.



(a) $d = 2$ (b) $d = 16$

Fig. 2: *Average* **ESS per minute of implementation** *for SUZZ(1), ZZ, RWM and IRWM algorithms in the logistic regression model for dimensions* $d = 2$ *and* $d = 16$ *and various number of observed data.*



(a) $d = 2$ (b) $d = 16$

Fig. 3: *Average* **ESS per likelihood evaluation** *for SUZZ(1), ZZ, RWM and IRWM algorithms in the logistic regression model for dimensions* $d = 2$ *and* $d = 16$ *and various number of observed data. The y-axis is in scale* 10^{-3}.

Finally, we should note that if one has an understanding of the posterior's covariance matrix (as is the case in logistic regression, used in the RWM case when tuning the proposal's covariance matrix), one can try to improve the performance of the original Zig-Zag and the SUZZ algorithm, for example using ideas from [2]. One could also try to use speed functions that take into account the known approximation C of the covariance matrix, for example by using

$$s(x) = \left(1 + x^T C^{-1} x\right)^{\frac{1+k}{2}} .$$

Using this type of ideas is work in progress.

Acknowledgements

We would like to thank Vasilis G. S. Vasdekis for pointing out results on the asymptotic covariance matrix of maximum likelihood estimators for the logistic regression. We would also like to thank the two anonymous referees for their helpful feedback. G. Vasdekis was supported by EPSRC (EP/V055380/1). G. O. Roberts was supported by EPSRC under the CoSInES (EP/R018561/1) and Bayes for Health (EP/R034710/1) programmes.

References

1. Andrieu, C., Dobson, P., Wang, A.Q.: Subgeometric hypocoercivity for piecewise-deterministic Markov process Monte Carlo methods. Electronic Journal of Probability **26**(none), 1 – 26 (2021). DOI 10.1214/21-EJP643
2. Bertazzi, A., Bierkens, J.: Adaptive schemes for piecewise deterministic Monte Carlo algorithms. Bernoulli **28**(4), 2404 – 2430 (2022). DOI 10.3150/21-BEJ1423
3. Bierkens, J.: Non-reversible metropolis-hastings. Statistics and Computing **26** (2014). DOI 10.1007/s11222-015-9598-x
4. Bierkens, J., Bouchard-Côté, A., Doucet, A., Duncan, A.B., Fearnhead, P., Lienart, T., Roberts, G.O., Vollmer, S.G.: Piecewise deterministic markov processes for scalable monte carlo on restricted domains. Statistics & Probability Letters **136**, 148 – 154 (2018). DOI https://doi.org/10.1016/j.spl.2018.02.021. URL http://www.sciencedirect.com/science/article/pii/S016771521830066X. The role of Statistics in the era of big data
5. Bierkens, J., Duncan, A.: Limit theorems for the zig-zag process. Advances in Applied Probability **49**(3), 791–825 (2017). DOI 10.1017/apr.2017.22
6. Bierkens, J., Fearnhead, P., Roberts, G.O.: The zig-zag process and super-efficient sampling for bayesian analysis of big data. Ann. Statist. **47**(3), 1288–1320 (2019). DOI 10.1214/18-AOS1715. URL https://doi.org/10.1214/18-AOS1715
7. Bierkens, J., Grazzi, S., Kamatani, K., Roberts, G.O.: The boomerang sampler. In: H.D. III, A. Singh (eds.) Proceedings of the 37th International Conference on Machine Learning, *Proceedings of Machine Learning Research*, vol. 119, pp. 908–918. PMLR, Virtual (2020). URL http://proceedings.mlr.press/v119/bierkens20a.html

8. Bierkens, J., Grazzi, S., van der Meulen, F., Schauer, M.: A piecewise deterministic monte carlo method for diffusion bridges. Statistics and Computing **31** (2021). DOI 10.1007/s11222-021-10008-8. URL https://doi.org/10.1007/s11222-021-10008-8
9. Bierkens, J., Nyquist, P., Schlottke, M.C.: Large deviations for the empirical measure of the zig-zag process. The Annals of Applied Probability **31**(6), 2811 – 2843 (2021). DOI 10.1214/21-AAP1663. URL https://doi.org/10.1214/21-AAP1663
10. Bierkens, J., Roberts, G.O., Zitt, P.A.: Ergodicity of the zigzag process. Ann. Appl. Probab. **29**(4), 2266–2301 (2019). DOI 10.1214/18-AAP1453. URL https://doi.org/10.1214/18-AAP1453
11. Bierkens, J., Verduyn Lunel, S.M.: Spectral analysis of the zigzag process (2019). To appear in *Annales de l'Institut Henri Poincaré (B) Probabilitès et Statistiques*. Available at https://arxiv.org/abs/1905.01691
12. Bouchard-Côté, A., Vollmer, S.J., Doucet, A.: The bouncy particle sampler: A nonreversible rejection-free markov chain monte carlo method. Journal of the American Statistical Association **113**(522), 855–867 (2018). DOI 10.1080/01621459.2017.1294075
13. Chen, T.L., Hwang, C.R.: Accelerating reversible markov chains. Statistics & Probability Letters **83**(9), 1956 – 1962 (2013). DOI https://doi.org/10.1016/j.spl.2013.05.002. URL http://www.sciencedirect.com/science/article/pii/S0167715213001533
14. Chopin, N., Ridgway, J.: Leave Pima Indians Alone: Binary Regression as a Benchmark for Bayesian Computation. Statistical Science **32**(1), 64 – 87 (2017). DOI 10.1214/16-STS581. URL https://doi.org/10.1214/16-STS581
15. Corbella, A., Spencer, S.E.F., Roberts, G.O.: Automatic zig-zag sampling in practice. Statistics and Computing **32** (2022). DOI 10.1007/s11222-022-10142-x
16. Cotter, S., House, T., Pagani, F.: The nuzz: Numerical zigzag sampling for general models (2020). Available in https://arxiv.org/abs/2003.03636
17. Diaconis, P., Holmes, S., Neal, R.M.: Analysis of a nonreversible Markov chain sampler. Ann. Appl. Probab. **10**(3), 726–752 (2000). DOI 10.1214/aoap/1019487508
18. Duncan, A., Lelièvre, T., Pavliotis, G.: Variance reduction using nonreversible langevin samplers. Journal of Statistical Physics **163** (2015). DOI 10.1007/s10955-016-1491-2
19. Fearnhead, P., Bierkens, J., Pollock, M., Roberts, G.O.: Piecewise deterministic Markov processes for continuous-time Monte Carlo. Statist. Sci. **33**(3), 386–412 (2018). DOI 10.1214/18-STS648
20. Hwang, C.R., Hwang-Ma, S.Y., Sheu, S.J.: Accelerating Gaussian diffusions. Ann. Appl. Probab. **3**(3), 897–913 (1993). DOI 10.1214/aoap/1177005371
21. Lelièvre, T., Nier, F., Pavliotis, G.A.: Optimal non-reversible linear drift for the convergence to equilibrium of a diffusion. Journal of Statistical Physics **152**(2), 237–274 (2013). DOI 10.1007/s10955-013-0769-x. URL https://hal.archives-ouvertes.fr/hal-00761688
22. McCullagh, P., Nelder, J.A.: Generalized Linear Models. 2nd Edition. Chapman and Hall, London (1989)
23. Roberts, G.O., Gelman, A., Gilks, W.R.: Weak convergence and optimal scaling of random walk metropolis algorithms. Ann. Appl. Probab. **7**(1), 110–120 (1997). DOI 10.1214/aoap/1034625254. URL https://doi.org/10.1214/aoap/1034625254
24. Sun, Y., Schmidhuber, J., Gomez, F.: Improving the asymptotic performance of markov chain monte-carlo by inserting vortices. In: J. Lafferty, C. Williams, J. Shawe-Taylor, R. Zemel, A. Culotta (eds.) Advances

in Neural Information Processing Systems, vol. 23, pp. 2235–2243. Curran Associates, Inc. (2010). URL https://proceedings.neurips.cc/paper/2010/file/819f46e52c25763a55cc642422644317-Paper.pdf
25. Sutton, M., Fearnhead, P.: Concave-convex pdmp-based sampling (2021). Available on https://arxiv.org/abs/2112.12897
26. Vanetti, P., Bouchard-Côté, A., Deligiannidis, G., Doucet, A.: Piecewise-deterministic markov chain monte carlo (2017). Available at https://arxiv.org/abs/1707.05296
27. Vasdekis, G., Roberts, G.O.: A note on the polynomial ergodicity of the one-dimensional zig-zag process. Journal of Applied Probability p. 1–9 (2022). DOI 10.1017/jpr.2021.97
28. Vasdekis, G., Roberts, G.O.: Speed Up Zig-Zag (2023+). Accepted in *Annals of Applied Probability*. Available at https://arxiv.org/abs/2103.16620