Numerische Mathematik (2020) 144:447–449
https://doi.org/10.1007/s00211-019-01086-y

Numerische Mathematik

CORRECTION

Correction to: Hybrid Monte Carlo methods for sampling probability measures on submanifolds

Tony Lelièvre[1] · Mathias Rousset[2] · Gabriel Stoltz[1]

Published online: 22 November 2019

Correction to: Numerische Mathematik (2019) 143:379–421 https://doi.org/10.1007/s00211-019-01056-4

Shiva Darshan and Miranda Holmes–Cerfon (Courant Institute, NYU) pointed out a mistake in the projection functions to enforce the momentum constraint when rewriting the algorithm in Numerical Algorithm A of Section 3.1. Two different projection functions are actually needed, see indeed the formula for the Lagrange multiplier λ^{n+1} after Equation (7) for the RATTLE step, and Remark 5 for the Ornstein–Uhlenbeck step.

We provide below a corrected version of the pseudo-code for the complete algorithm; see Numerical algorithms 1, 2 and 3. Changes are highlighted in blue. The sampling algorithm consists in iterating procedure `ConstrainedGHMC` of the algorithm (Numerical algorithm 1), which uses the procedures `LAGRANGE_MOMENTUM_OU` (Numerical algorithm 2) and `LAGRANGE_MOMENTUM_RATTLE` (Numerical algorithm 3) to compute the Lagrange multiplier for momentum constraints in the fluctuation/dissipation and RATTLE steps, respectively. The procedure `NEWTON` to compute the Lagrange multiplier for position constraints is unchanged.

Numerical algorithms 2 and 3 differ by a multiplication by $\left(\mathrm{Id} + \Delta t \gamma M^{-1}/4\right)^{-1}$, which arises from the specific choice of the discretization of the fluctuation/dissipation part in Algorithm 3.

The original article can be found online at https://doi.org/10.1007/s00211-019-01056-4.

✉ Gabriel Stoltz
gabriel.stoltz@enpc.fr

1 CERMICS (ENPC), Inria, Université Paris-Est, 77455 Marne-la-Vallée, France

2 SIMSART Team-Project, Inria Rennes, France

Numerical algorithm 1 One step of the practical constrained HMC algorithm with reverse projection

Parameters: γ (friction), Δt (timestep), η_{rev} (tolerance for reverse check)

```
procedure ConstrainedGHMC(q,p)
  G ~ N(0, Id)
  p ← (Id + Δt γ M^{-1}/4)^{-1} [(Id − Δt γ M^{-1}/4) p + √(γΔt) G]
  λ = LAGRANGE_MOMENTUM_OU(q, p)
  p ← p + (Id + Δt γ M^{-1}/4)^{-1} ∇ξ(q) λ          ▷ Integration of the fluctuation/dissipation for Δt/2
  Reject = TRUE
  p̃ = p − Δt ∇V(q)/2 and q̃ = q + Δt M^{-1} p̃
  Compute (Success_forward_RATTLE, θ) = NEWTON(q, q̃)
  if Success_forward_RATTLE then
    p̃ ← p̃ + ∇ξ(q)θ/Δt and q̃ ← q̃ + M^{-1} ∇ξ(q)θ
    p̃ ← p̃ − Δt ∇V(q̃)/2
    λ = LAGRANGE_MOMENTUM_RATTLE(q̃, p̃)
    p̃ ← p̃ + ∇ξ(q̃)λ                                  ▷ Constrained RATTLE – proposition
    p̂ = −p̃ and q̂ = q̃
    p̂ ← p̂ − Δt ∇V(q̂)/2 and q̂ ← q̂ + Δt M^{-1} p̂
    Compute (Success_backward_RATTLE, θ) = NEWTON(q̃, q̂)
    if Success_backward_RATTLE then
      p̂ ← p̂ + ∇ξ(q̃)θ/Δt and q̂ ← q̂ + M^{-1} ∇ξ(q̃)θ
      p̂ ← p̂ − Δt ∇V(q̂)/2
      λ = LAGRANGE_MOMENTUM_RATTLE(q̂, p̂)
      p̂ ← p̂ + ∇ξ(q̂)λ                                ▷ Constrained RATTLE – reverse move
      if ‖q̂ − q‖ < η_rev then                        ▷ Constrained RATTLE – checking reversibility
        U ~ U([0, 1])
        ΔH = H(q̃, p̃) − H(q, p)
        if log(U) ≤ −ΔH then                         ▷ Constrained RATTLE – Metropolis acceptance/rejection
          Reject = FALSE
        end if
      end if
    end if
  end if
  if Reject then
    p̃ = −p and q̃ = q
  end if
  G̃ ~ N(0, Id)
  p̃ ← (Id + Δt γ M^{-1}/4)^{-1} [(Id − Δt γ M^{-1}/4) p̃ + √(γΔt) G̃]
  λ = LAGRANGE_MOMENTUM_OU(q̃, p̃)
  p̃ ← p̃ + (Id + Δt γ M^{-1}/4)^{-1} ∇ξ(q̃) λ          ▷ Integration of the fluctuation/dissipation for Δt/2
  return q̃, p̃
end procedure
```

Numerical algorithm 2 Computation of the Lagrange multiplier for momentum constraints in OU part

```
procedure LAGRANGE_MOMENTUM_OU(q,p)
  S = [∇ξ(q)]^T M^{-1} (Id + Δt γ M^{-1}/4)^{-1} ∇ξ(q)
  b = [∇ξ(q)]^T M^{-1} p
  return λ = −S^{-1} b
end procedure
```

Numerical algorithm 3 Computation of the Lagrange multiplier for momentum constraints in RATTLE part

procedure LAGRANGE_MOMENTUM_RATTLE(q,p)
  $S = [\nabla\xi(q)]^T\ M^{-1}\nabla\xi(q)$
  $b = [\nabla\xi(q)]^T\ M^{-1}p$
  return $\lambda = -S^{-1}b$
end procedure