Preliminary results
Discrete PDMP in BAT

Athina Monemvassitis

4, May 2022

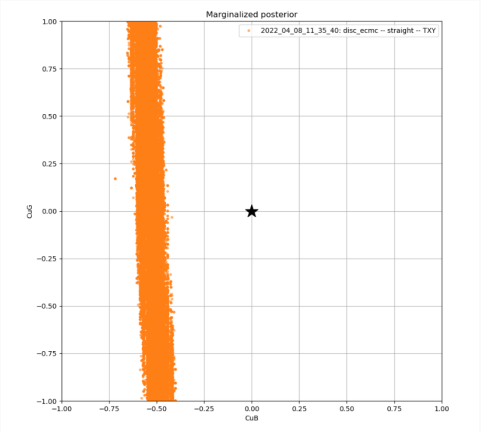
athina.monemvassitis@uca.fr

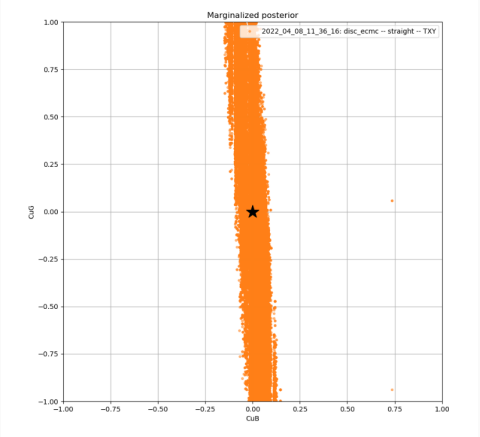
Minutes

4, May 2022, online Zoom meeting

Sampling Cornelius' thesis distribution

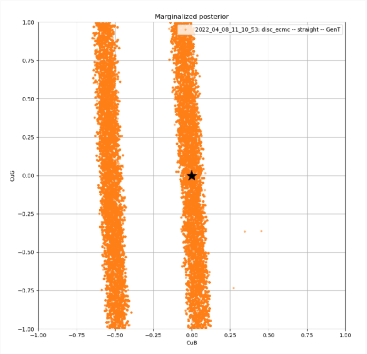
Sampling Cornelius' thesis distribution – XY Translations

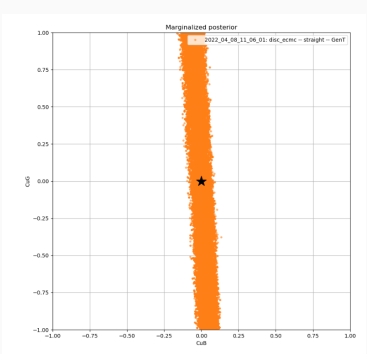


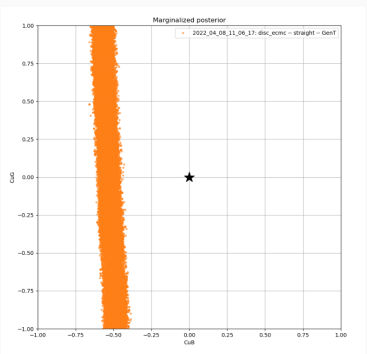


CuB VS CuG, Two different runs with random initial conditions

Sampling Cornelius' thesis distribution – General Translations

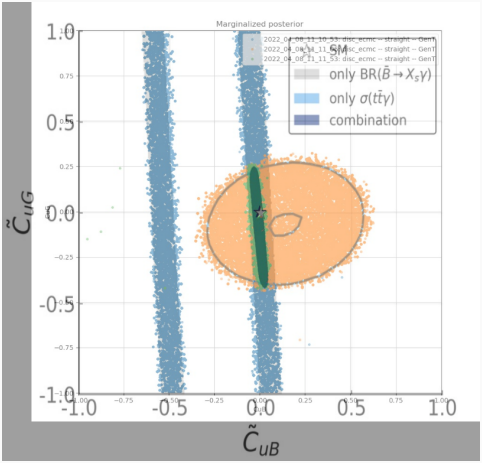






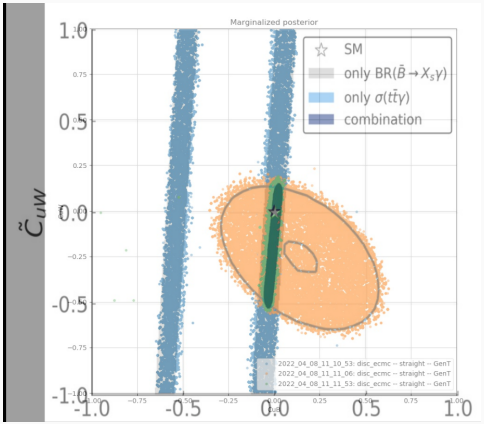
CuB VS CuG, three different runs with $\langle \delta \rangle = 0.5, 0.05, 0.05$

Sampling Cornelius' thesis distribution – General Translations



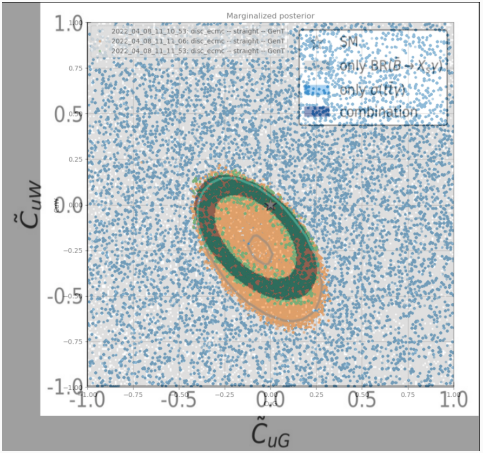
CuB VS CuG, for $\langle\delta\rangle = 0.5$

Sampling Cornelius' thesis distribution – General Translations



CuB VS CuW for $\langle\delta\rangle = 0.5$

Sampling Cornelius' thesis distribution – General Translations



CuG VS CuW for $\langle\delta\rangle = 0.5$

Conclusion section

- Apparently good agreement
  - Metodology for the selection of the points (90%)
  - Experimental data points from the paper for comparison
- (Lars and I ?) Tuning the algorithm
  - Include it in the BAT software ? (see next section)
  - Tune the step size analogously to the Metropolis sampler ?
- (all ?) Efficiency estimation
  - Done directly in BAT ?
  - BAT tools include decorrelation calculations ?
  - Definition of good observables / estimators hard to deorrelate
  - Discuss about this together with Manon ?

    → Monday 9, May 2022 – 11h-12h

    → Monday 9, May 2022 – 12h-13h

Structure of the Discrete PDMP algorithm

Structure of the Discrete PDMP algorithm

Discrete PDMP for 1 new sample

Require: $x \in \mathcal{X}$, $v \in \mathcal{V}$, $\delta > 0$, number_steps > 0
Ensure: New sample $y \in \mathcal{X}$
  for step in number_step **do**
    $x' =$ propose_move(x, v, δ)
    $p_{\text{acc}} =$ get_pacc(x, x')
    if p_{acc} **then**
      $x \leftarrow x'$
    else
      $v =$ update_lift_variable(x, v, δ)
    end if
  end for

- Tunning parameters
  - step size δ for acceptance rate
  - number_steps between two samples for decorrelation
- Specified by the sampling scheme only
  - v (example in 2D) :
    - → XY translations : $v \in \{(1,0),(0,1)\}$
    - → General translations : $v \in \{\mathbb{S}_2\}$
- Specified by the sampling scheme or by the distribution
  - get_pacc
    - → $\Delta E(x)$, $\nabla E(x)$
  - propose_move
    - → might depend on v
    - → might depend on the state space (periodic boundary conditions, ...)
  - update_lift_variable
    - → depends on the chosen scheme through v
    - → $\Delta E(x)$, $\nabla E(x)$ for changes in v

Structure of the Discrete PDMP algorithm – BAT ?

Discrete PDMP – in Distribution file

procedure $\mathrm{E}((x))$
    # Computation of the Energy function
    return Energy
end procedure
procedure GRAD_$\mathrm{E}((x))$
    # Computation of the Gradient Energy function
    return Gradient Energy
end procedure
procedure PROPOSE_MOVE_$$\mathrm{SCHEME}((x, v, \delta))$$
    # Implementation of the move according to _$SCHEME
    return a new position x'
end procedure

Discrete PDMP – in Distribution file

procedure GET_PACC$((x, x'))$
    # Implementation of the acceptance probability
    acc = False
    $p = \min(1, \exp(-(E(x') - E(x))))$
    if rand(0,1) $< p$ **then**
        acc = True
    end if
    return acc
end procedure
procedure UPDATE_VARIABLE_$$\mathrm{SCHEME}((x))$$
    # Updates v variable according to $SCHEME
    return v
end procedure

Structure of the Discrete PDMP algorithm – Schemes ?

Dependence between chosen scheme and distribution

- Irreversible (what I used for Cornelius' distribution)
  - Possible to code it such that only the energy difference needs to be specified by the user
  - Loss of the benefits of XY translations in the case of independ marginals
- Forward
  - If added, needs in addition the gradient of the energy

Possibility – In Sampler file

- Implement all the possible propose_move functions for the possible schemes
- Implement all the possible update_variable functions for the possible schemes
- Implement get_pacc for energy difference in the case of XY translations or General translations (To get faster sampling in the case of independent marginals)

$\Longrightarrow$ (for the user) Total energy difference, Total gradient energy

$\Longrightarrow$ (for the user) Marginal energy difference, Marginal gradient energy in the case of independent marginals for the Forward XY translations scheme

Conclusion section

- (Lars) Structure OK for the BAT software ?
- Possibility to encode for easy use :

  the user needs only to specify
  - The scheme
    - $\rightarrow$ XY translation or General translations
    - $\rightarrow$ Irreversible or forward updates (there are other that can be implemeted as easily)
  - The tuning parameters if not tuned in BAT (δ, number of steps between two samples, number of steps between two refreshments)

  as well as implementing
  - The energy difference
  - The gradient of the energy if scheme forward (or another to be implemented) selected
- Note : holds for the most of the distributions I can think about but not for spheres

  (I can develop if needed)