Divide-and-Conquer Generalised Bayesian Fusion
A method of unifying distributed analyses

Ryan Chan [1,3], Murray Pollock [1,2], Adam Johansen [1,3], Gareth Roberts [1,3]

[1] University of Warwick, [2] Newcastle University, [3] The Alan Turing Institute

Generalised Monte Carlo Fusion (GMCF)

Provides theory to carry out *exact* inference for the target:

$$f(\boldsymbol{x}) \propto f_1(\boldsymbol{x}) \cdots f_C(\boldsymbol{x}) = \prod_{c=1}^{C} f_c(\boldsymbol{x}). \tag{1}$$

Uses *importance sampling* on the extended target distribution:

$$g\left(\boldsymbol{x}^{(1:C)}, \boldsymbol{y}\right) \propto \prod_{c=1}^{C} \left[f_c^2\left(\boldsymbol{x}^{(c)}\right) \cdot p_c\left(\boldsymbol{y} \middle| \boldsymbol{x}^{(c)}\right) \cdot \frac{1}{f_c\left(\boldsymbol{y}\right)} \right]. \tag{2}$$

Let $p_c(\boldsymbol{y} \mid \boldsymbol{x}^{(c)})$ be a transition density of a stochastic process with stationary distribution $f_c^2(\boldsymbol{x})$, then (2) admits f as a marginal for $\boldsymbol{y}$.
There are many possible choices for p_c, but we focus on letting p_c be transition density of a d-dimensional double Langevin diffusion process for $t \in [0, T]$:

$$\mathrm{d}\boldsymbol{X}_t^{(c)} = \boldsymbol{\Lambda}_c \nabla \log f_c\left(\boldsymbol{X}_t^{(c)}\right) \mathrm{d}t + \boldsymbol{\Lambda}_c^{\frac{1}{2}} \mathrm{d}\boldsymbol{W}_t^{(c)}, \quad \boldsymbol{X}_0^{(c)} = \boldsymbol{x}^{(c)}, \ \boldsymbol{X}_T^{(c)} = \boldsymbol{y}. \tag{3}$$

Proposal density:

$$h\left(\boldsymbol{x}^{(1:C)}, \boldsymbol{y}\right) \propto \prod_{c=1}^{C} \left[f_c\left(\boldsymbol{x}^{(c)}\right) \right] \cdot \exp\left\{ -\frac{(\boldsymbol{y} - \tilde{\boldsymbol{x}})^\intercal \boldsymbol{\Lambda}^{-1} (\boldsymbol{y} - \tilde{\boldsymbol{x}})}{2T} \right\}, \tag{4}$$

where

$$\tilde{\boldsymbol{x}} := \left(\sum_{c=1}^{C} \boldsymbol{\Lambda}_c^{-1} \right)^{-1} \left(\sum_{c=1}^{C} \boldsymbol{\Lambda}_c^{-1} \boldsymbol{x}^{(c)} \right), \quad \boldsymbol{\Lambda}^{-1} := \sum_{c=1}^{C} \boldsymbol{\Lambda}_c^{-1}. \tag{5}$$

Under certain mild conditions,

$$\frac{g\left(\boldsymbol{x}^{(1:C)}, \boldsymbol{y}\right)}{h\left(\boldsymbol{x}^{(1:C)}, \boldsymbol{y}\right)} \propto \rho_0 \cdot \rho_1 =: w'\left(\boldsymbol{x}^{(1:C)}, \boldsymbol{y}\right), \tag{6}$$

where

$$\rho_0\left(\boldsymbol{x}^{(1:C)}\right) := \exp\left\{ -\sum_{c=1}^{C} \frac{(\tilde{\boldsymbol{x}} - \boldsymbol{x}^{(c)})^\intercal \boldsymbol{\Lambda}_c^{-1} (\tilde{\boldsymbol{x}} - \boldsymbol{x}^{(c)})}{2T} \right\}, \tag{7}$$

$$\rho_1\left(\boldsymbol{x}^{(1:C)}, \boldsymbol{y}\right) := \prod_{c=1}^{C} \mathbb{E}_{\mathbb{W}_{\boldsymbol{\Lambda}_c}} \left[\exp\left\{ -\int_0^T \phi_c\left(\boldsymbol{X}_t^{(c)}\right) \mathrm{d}t \right\} \right], \tag{8}$$

and $\phi_c(\boldsymbol{x}) := \frac{1}{2}\left(\nabla \log f_c(\boldsymbol{x})^\intercal \boldsymbol{\Lambda}_c \nabla \log f_c(\boldsymbol{x}) + \mathrm{Tr}\left(\boldsymbol{\Lambda}_c \nabla^2 \log f_c(\boldsymbol{x})\right)\right)$, where $\mathbb{W}_{\boldsymbol{\Lambda}_c}$ denotes the law of a Brownian bridge $\{\boldsymbol{X}_t^{(c)}, t \in [0, T]\}$ with $\boldsymbol{X}_0^{(c)} := \boldsymbol{x}^{(c)}$, $\boldsymbol{X}_T^{(c)} := \boldsymbol{y}$ and covariance matrix $\boldsymbol{\Lambda}_c$.

Algorithm 1 Generalised Monte Carlo Fusion (GMCF)

1. Simulate a proposal $\boldsymbol{y}$ from h (4):
   a) For $c = 1, \ldots, C$, simulate $\boldsymbol{x}^{(c)} \sim f_c(\boldsymbol{x})$ and calculate $\tilde{\boldsymbol{x}}$.
   b) Simulate $\boldsymbol{y} \sim \mathcal{N}_d(\tilde{\boldsymbol{x}}, T\boldsymbol{\Lambda})$.
2. Assign unnormalised weight $w'\left(\boldsymbol{x}^{(1:C)}, \boldsymbol{y}\right)$ as per (6).

Interpretation: GMCF corrects a perturbation of a weighted average of Monte Carlo samples from the sub-posteriors.

Generalised Bayesian Fusion (GBF)

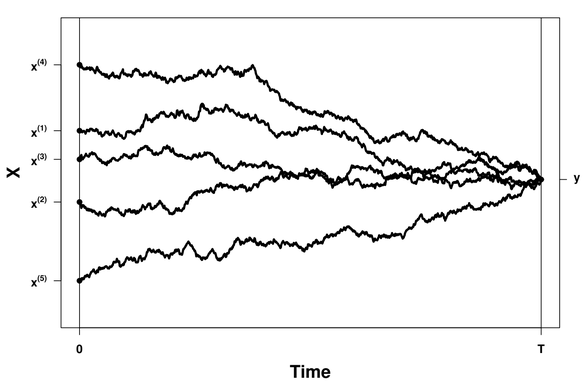


Figure 1: GMCF approach.

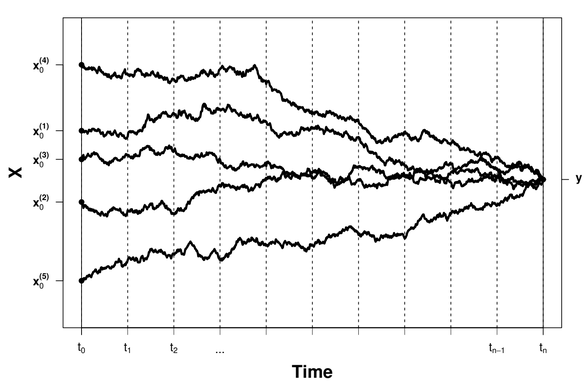


Figure 2: GBF approach.

- GMCF simulates C independent stochastic processes such that they coalesce at time T and the marginal distribution at time T has density f (1).
- GBF replaces IS with a sequential Monte Carlo approach which steps through a sequence of distributions between the initial proposal distribution to the target fusion density by considering a temporal partition of $[0, T]$: $\mathcal{P} = \{t_0, t_1, \ldots, t_n : 0 =: t_0 < t_1 < \cdots < t_n := T\}$.

Divide-and-Conquer Fusion

Problem: Fusion becomes inefficient as the number of sub-posteriors increases.

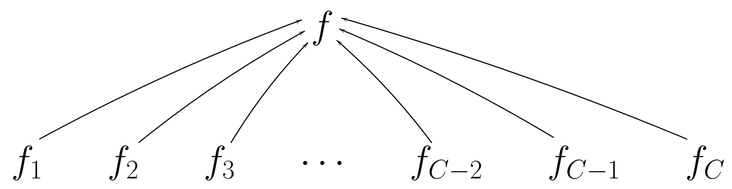


Figure 3: The 'fork-and-join' approach.

However, we can adopt a *divide-and-conquer* approach and embed our Fusion algorithm within a *divide-and-conquer sequential Monte Carlo* framework:

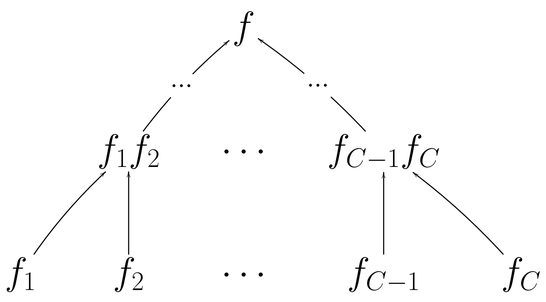


Figure 4: A balanced-binary tree.

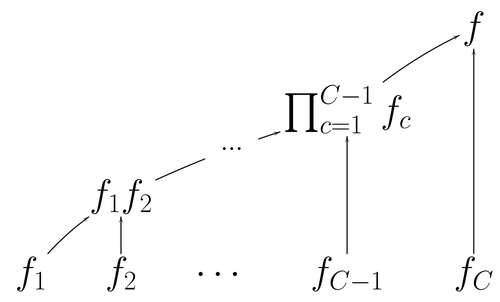
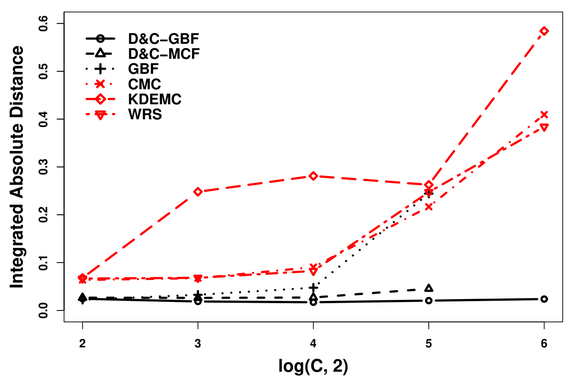


Figure 5: A progressive tree.

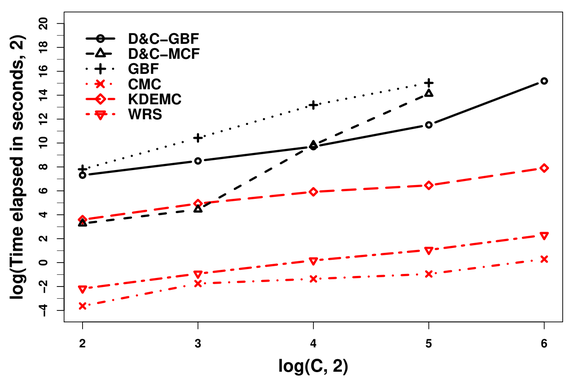
Logistic regression

- Simulated data with $n = 1000$ and $d = 5$
- To compare methods, use *IAD* (where $f(\boldsymbol{x}_j)$ is target density) (lower better):

$$IAD = \frac{1}{2d} \sum_{j=1}^{d} \int \left| \hat{f}(\boldsymbol{x}_j) - f(\boldsymbol{x}_j) \right| \mathrm{d}x_j \in [0, 1].$$



(a): Integrated Absolute Distance.

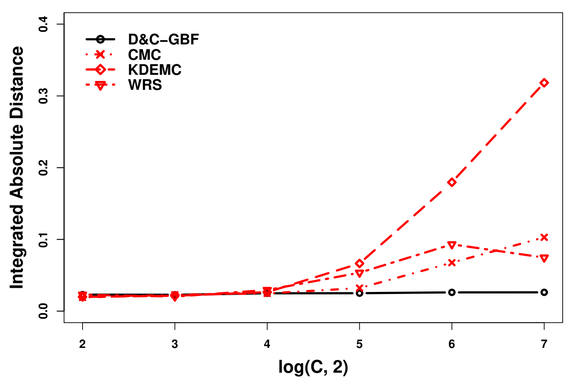


(b): Computational cost.

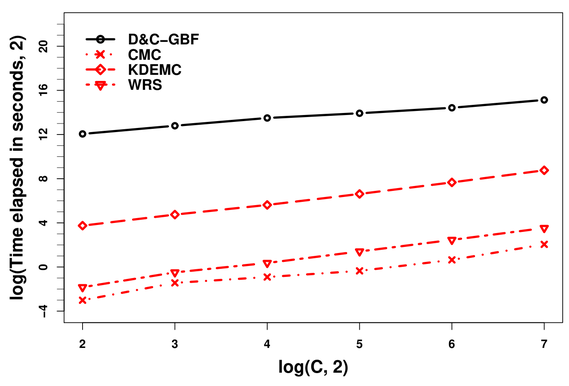
Figure 6: Comparison of competing methodologies to Fusion applied to logistic regression.

Negative Binomial regression

- Bike sharing dataset ($n = 17379$, $d = 10$)



(a): Integrated Absolute Distance.



(b): Computational cost.

Figure 7: Comparison of competing methodologies to Fusion applied to NB regression.

References

- Chan, R.S.Y., Johansen, A.M., Pollock, M., and Roberts, G.O. 2021. *Divide-and-Conquer Monte Carlo Fusion.* Submitted. arXiv:2110.07265.
- Dai, Hongsheng, Murray Pollock, and Gareth Roberts. *Monte Carlo Fusion.* Journal of Applied Probability (56)1. 2019: pp. 174-191.
- Dai, H., Pollock, M. and Roberts, G., 2021. *Bayesian Fusion: Scalable unification of distributed statistical analyses.* Submitted. arXiv:2102.02123.