Generative Bayesian Computation for Probabilistic Forecasting with Discrete Events

Anonymous Authors[1]

Abstract

Many forecasting problems feature events with known timing but unknown magnitude: earnings announcements, central bank decisions, and election dates are all scheduled months in advance, yet standard models treat event timing as a latent Bernoulli draw, discarding this calendar information. We develop sequential Generative Bayesian Computation (GBC) as an online learning framework grounded in proper scoring rules: draw candidates from a tractable generative model, an implicit quantile network, reweight by a scoring-rule loss, and resample; known event dates condition the generator directly, eliminating latent-variable timing inference. For financial returns, this yields an earnings-jump model with an analytic Gaussian jump-size posterior on announcement days and no ESS collapse. For stochastic volatility, the mixture Kalman filter is the optimal GBC generator, raising effective sample size from 0.73 to 0.92 while maintaining calibrated 90% credible coverage, as confirmed across 100 replications.

1. Introduction

Earnings announcements, central bank rate decisions, employment releases, and election outcomes all share a structural property: the event will happen on a known date, but its magnitude or direction is uncertain until it does. Despite this, the dominant modelling approach treats event timing as a latent Bernoulli draw at every time step (Eraker et al., 2003), turning a known calendar date into an inference problem.

Exploiting known event timing requires a framework where the generator can condition on observable schedule information. We develop *sequential Generative Bayesian Computation (GBC)* for this purpose. GBC recasts probabilistic forecasting as online learning with proper scoring rules (Gneiting & Raftery, 2007): draw state candidates from a generative model, the implicit quantile network (Dabney et al., 2018), reweight by a proper-scoring-rule loss, and resample. This is a decision-theoretically coherent Bayesian update (Bissiri et al., 2016) expressed through scores, making calibration a consequence of generator design. Known event dates enter the generator as inputs rather than latent variables to be inferred.

Sequential GBC rests on a *noise outsourcing* result: the filtering distribution is a recursive composition of conditional quantile maps covering discrete, deterministic, and stochastic dynamics without density evaluation. Better generators, measured by chi-squared divergence to the optimal map, yield higher ESS and better-calibrated predictives.

The paper makes three contributions. (1) We establish sequential GBC as a scoring-rule-based framework for probabilistic forecasting, with deterministic event handling as a first-class feature. (2) We introduce the deterministic-timing earnings-jump model, where event dates are observed rather than inferred, and derive an exact Gaussian update for jump-size inference on announcement days. (3) We show that the mixture Kalman filter is the optimal GBC generator for the KSC-linearized stochastic volatility model, and quantify its ESS advantage over the bootstrap particle filter across 100 replications.

Related work. Neural forecasting (foundation models Chronos (Ansari et al., 2024) and TimesFM (Das et al., 2024)) and SMC methods (Gordon et al., 1993; Carvalho et al., 2010a) treat event timing as latent or ignore calendar information. Proper scoring rules (Gneiting & Raftery, 2007; Bissiri et al., 2016) ground the GBC weighting step; the discrete-events principle is orthogonal, making event dates inputs rather than latent variables.

2. Sequential GBC for Discrete-Event Forecasting

We develop sequential GBC for dynamic settings by constructing the filtering distribution as a recursive composition of conditional quantile maps.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

A state space model (SSM) specifies latent state $x_t \in \mathcal{X}$ and observation $y_t \in \mathcal{Y}$ with Markov dynamics $p(x_t \mid x_{t-1})$ and emission $p(y_t \mid x_t)$. The filtering distribution is $\pi_t = \mathcal{L}(x_t \mid y_{1:t})$.

Proposition 1 (Sequential Noise Outsourcing)**.** *Let $\mathcal{X}$ be a standard Borel space and the kernels measurable. There exist i.i.d. $U_t \sim \mathrm{Uniform}(0,1)$ and measurable maps $F_t : [0,1] \times \mathcal{X} \times \mathcal{Y}^t \to \mathcal{X}$ such that $\hat{x}_t = F_t(U_t, \hat{x}_{t-1}, y_{1:t})$ satisfies $\mathcal{L}(\hat{x}_t \mid y_{1:t}) = \pi_t$ for every t. For $\mathcal{X} \subseteq \mathbb{R}$, F_t is the conditional quantile function, unique up to a null set.*

Ionescu-Tulcea builds the path-space product measure; the Markov property factors it into one-step kernels. Each $F_t(u; x_{t-1}, y_{1:t}) = \inf\{x : \pi_t((-\infty, x] \mid x_{t-1}, y_{1:t}) \geq u\}$ is the Parzen conditional quantile function (Parzen, 2004): the generalised inverse CDF, well-defined for continuous, discrete, or mixed distributions, covering the KSC indicator r_t, regime variable s_t, and jump indicator d_t (Carlin & Polson, 1991; Carlin et al., 1992). The non-trivial step was showing F_t depends on $x_{t-1} = F_{t-1}(U_{t-1}, \ldots)$: the full trajectory is the sequential composition $F = F_T \circ \cdots \circ F_1$ driven by i.i.d. uniforms.

Definition 1 (GBC Algorithm)**.** *A GBC algorithm with N particles approximates π_t by: (a) drawing candidates $x_t^{(i)} = \hat{F}_t(u^{(i)}, x_{t-1}^{(i)}, y_{1:t})$ from a tractable approximation $\hat{F}_t \approx F_t$; (b) weighting $w_t^{(i)} = \exp(-\ell(x_t^{(i)}; y_t))$ for a proper-scoring-rule loss ℓ; (c) resampling proportionally to $\{w_t^{(i)}\}$.*

Better generators yield smaller chi-squared divergence $\chi^2(F_t \| \hat{F}_t)$ and therefore higher effective sample size (ESS); from importance-sampling theory: $\mathrm{ESS}_t \approx N/(1 + \chi^2(F_t \| \hat{F}_t))$.

The generator $\hat{F}_t$ approximates F_t; an IQN (Dabney et al., 2018) trained on the pinball loss provides a natural parametrisation when no analytic structure is available. The loss ℓ spans the proper-scoring-rule class (Gneiting & Raftery, 2007): the log-score recovers likelihood weighting; the CRPS is a density-free alternative.

Earnings announcement dates fix jump timing but not magnitude; FOMC calendars reveal meeting days but not rate decisions; election days are set while vote margins are not. In each case the forecaster observes $d_t = 1\{t \in \mathcal{E}\}$ from a public calendar; only the event magnitude Z_t is latent.

Applied jump models in the Merton tradition (Merton, 1976; Eraker et al., 2003) treat the event indicator d_t as a draw from $\mathrm{Ber}(\lambda)$ at every step, even when the scheduling information is available. This creates two inefficiencies: the filter spends particles inferring something already known, and particles that drew the wrong timing receive near-zero weights on event days, collapsing the ESS.

Sequential GBC resolves this by design. Since d_t is observed, the generator conditions on it:

$$x_t^{(i)} = \hat{F}_t(u^{(i)},\, x_{t-1}^{(i)},\, y_{1:t-1},\, d_t). \tag{1}$$

The event indicator enters as input to the generator, not as a latent variable. On event days ($d_t = 1$), the generator additionally draws the event magnitude Z_t from its prior; the scoring-rule weight then updates this draw, giving an exact posterior for Z_t in conjugate settings without importance resampling.

3. Financial Applications

We instantiate sequential GBC for stochastic volatility (§3.1) and earnings jumps with deterministic timing (§3.2).

3.1. Stochastic Volatility via MKF-GBC

The canonical SV model has log-returns y_t and log-volatility x_t:

$$x_t = \phi x_{t-1} + \sigma \eta_t, \qquad y_t = \beta e^{x_t/2} \varepsilon_t, \tag{2}$$

with $\eta_t, \varepsilon_t \overset{\text{iid}}{\sim} \mathcal{N}(0,1)$. The Kim–Shephard–Chib (KSC) $K = 7$ mixture approximation to $\log \chi^2(1)$ (Kim et al., 1998) converts the SV model into a conditionally linear Gaussian SSM with discrete indicator $r_t \in \{1, \ldots, 7\}$. The mixture Kalman filter (MKF) (Chen & Liu, 2000) is the optimal GBC generator for the KSC-linearized model: conditioning on $r_t^{(i)}$ makes each component exactly linear Gaussian, so the Kalman update achieves zero chi-squared divergence within that component. The sufficient statistic for each particle is $(r_t^{(i)}, m_t^{(i)}, P_t^{(i)})$: the mixture indicator plus the conditional Kalman mean and variance. Adding a jump appends $Z_t^{(i)}$ from (6) to the sufficient statistic on event days; observing d_t lets the particle cloud discriminate vol-regime changes from earnings surprises, a confusion unavoidable when timing is latent (Carvalho et al., 2010b).

3.2. Earnings Jumps with Known Timing

We model log-returns on individual stocks as

$$x_t = \phi x_{t-1} + \sigma \eta_t, \tag{3}$$

$$y_t = \mu + e^{x_t/2} \varepsilon_t + d_t Z_t, \tag{4}$$

$$Z_t \sim \mathcal{N}(\mu_J, \sigma_J^2) \quad \text{when } d_t = 1, \tag{5}$$

where $d_t = 1\{t \in \mathcal{E}\}$ is from the earnings calendar and Z_t is the latent surprise; Dubinsky et al. (2019) document this structure empirically for Intel (INTC) and five other large-cap stocks. Table 1 contrasts this with the Merton model.

Table 1. Stochastic vs. deterministic jump timing.

Aspect	Merton (J_t latent)	Earnings (d_t observed)
Latent state	(x_t, J_t, Z_t)	(x_t, Z_t) on $d_t = 1$
Timing inference	Bernoulli at each step	Not needed
Magnitude update	Mixture of 2 terms	Exact Gaussian
ESS on event days	Collapses	Unaffected

On non-announcement days ($d_t = 0$), MKF-GBC from Section 3.1 applies directly. On announcement days ($d_t = 1$), particle i evaluates the marginal observation weight $\mathcal{N}(y_t;\, \mu + \mu_J,\, e^{x_t^{(i)}} + \sigma_J^2)$ exactly; no KSC step is needed because conditioning on $x_t^{(i)}$ gives an exact Gaussian likelihood. After resampling, the jump-size posterior is the analytic Gaussian:

$$Z_t \mid x_t^{(i)}, y_t \sim \mathcal{N}\left(\frac{\sigma_J^2 (y_t - \mu) + e^{x_t^{(i)}} \mu_J}{e^{x_t^{(i)}} + \sigma_J^2},\, \frac{e^{x_t^{(i)}} \sigma_J^2}{e^{x_t^{(i)}} + \sigma_J^2} \right). \tag{6}$$

No importance resampling is required for Z_t: the conjugate update is exact.

4. Simulation Study

We evaluate MKF-GBC on the canonical SV benchmark of Kim et al. (1998) with $\phi = 0.97$, $\sigma = 0.15$, $\beta = 1$, using $T = 500$ observations and $N = 1000$ particles, averaged over 100 independent replications. Three methods are compared: the bootstrap particle filter (BPF) (Gordon et al., 1993), a likelihood-tempered GBC baseline (LTGBC, tempering $\alpha = 0.5$), and MKF-GBC. Table 2 reports RMSE, ESS $/N$, and empirical 90% credible-interval coverage.

Table 2. SV results: 100 replications, $T = 500$, $N = 1000$. Means; Monte Carlo SE ≤ 0.005 throughout.

Model	Method	RMSE	ESS$/N$	90% Cov.
SV	BPF	0.405	0.732	0.896
	LTGBC	0.406	0.950	0.893
	MKF-GBC	0.405	0.916	0.895
Regime	MKF-GBC	0.933	0.669	0.896

MKF-GBC raises ESS $/N$ from 0.732 to 0.916 at no cost in RMSE or coverage: the conditional Kalman update is closer to the optimal generator than the BPF. LTGBC achieves 0.950 by likelihood tempering but requires an extra reweighting step. Figure 1 shows representative ESS traces and filtered log-volatility with credible bands.

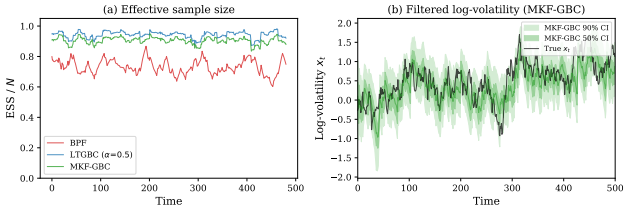

Figure 1. SV benchmark ($\phi = 0.97$, $\sigma = 0.15$, $\beta = 1$, $N = 1000$). *Left*: 20-step moving-average ESS $/N$ for BPF (blue), LTGBC (orange), and MKF-GBC (green). *Right*: filtered log-volatility from MKF-GBC; posterior median with 50% and 90% credible bands vs. true state. One replication shown; aggregates in Table 2.

We also evaluate MKF-GBC on a two-regime structural break model (Table 2 bottom row; Figure 2) with $K = 2$ regimes ($\phi_1 = 0.9, \phi_2 = 0.5$, $Q_1 = Q_2 = 1$, $R_1 = 1$, $R_2 = 4$, $\pi_{11} = \pi_{22} = 0.98$). The filter tracks regime switches sequentially; the sufficient statistic for each particle is $(s_t^{(i)}, m_t^{(i)}, P_t^{(i)})$, the discrete regime indicator plus the conditional Kalman moments. MKF-GBC achieves 90% credible coverage while maintaining ESS $/N = 0.669$. No BPF comparison is included; our KSC-based implementation was tailored to the SV structure and was not adapted to this model.

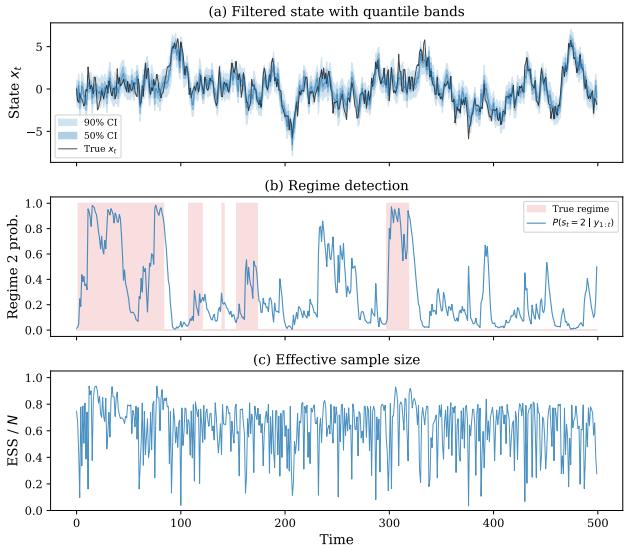

Figure 2. Structural break model ($K = 2$ regimes, $N = 1000$). *Left*: filtered state with 50%/90% bands. *Centre*: posterior regime probability $P(s_t = 2 \mid y_{1:t})$ vs. true regime (shaded). *Right*: ESS over time. One replication; aggregates in Table 2.

For the earnings-jump model, the gain is analytic: each particle carries an exact Gaussian posterior for Z_t on announcement days (6), eliminating the ESS collapse the Merton filter incurs when jump timing is latent.

5. Discussion

Sequential GBC unifies probabilistic forecasting under proper scoring rules: generator quality determines both ESS and calibration, and the discrete-events principle applies wherever event calendars are public.

The general loss in GBC (Bissiri et al., 2016) permits the energy score or CRPS for fat-tailed or partially specified models. Learned generators (IQN (Dabney et al., 2018), Mamba (Gu & Dao, 2023), or conditional diffusion models) parametrise $\hat{F}_t$ via pinball loss, selective state space, or reverse denoising chain when no analytic structure is available.

References

Ansari, A. F., Stella, L., Türkmen, C., et al. Chronos: Learning the language of time series. *Transactions on Machine Learning Research*, 2024.

Bissiri, P. G., Holmes, C. C., and Walker, S. G. A general framework for updating belief distributions. *Journal of the Royal Statistical Society: Series B*, 78(5):1103–1130, 2016.

Carlin, B. P. and Polson, N. G. Inference for nonconjugate Bayesian models using the Gibbs sampler. *Canadian Journal of Statistics*, 19(4):399–405, 1991.

Carlin, B. P., Polson, N. G., and Stoffer, D. S. A Monte Carlo approach to nonnormal and nonlinear state-space modeling. *Journal of the American Statistical Association*, 87(418):493–500, 1992.

Carvalho, C. M., Johannes, M. S., Lopes, H. F., and Polson, N. G. Particle learning and smoothing. *Statistical Science*, 25(1):88–106, 2010a.

Carvalho, C. M., Lopes, H. F., Polson, N. G., and Taddy, M. A. Particle learning for general mixtures. *Bayesian Analysis*, 5(4):709–740, 2010b.

Chen, R. and Liu, J. S. Mixture kalman filters. *Journal of the Royal Statistical Society: Series B*, 62(3):493–508, 2000.

Dabney, W., Ostrovski, G., Silver, D., and Munos, R. Implicit quantile networks for distributional reinforcement learning. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, 2018.

Das, A., Kong, W., Sen, R., and Zhou, Y. A decoder-only foundation model for time-series forecasting. In *Proceedings of the 41st International Conference on Machine Learning*, Proceedings of Machine Learning Research, 2024.

Dubinsky, A., Johannes, M., Kaeck, A., and Seeger, N. J. Option pricing of earnings announcement risks. *Review of Financial Studies*, 32(2):646–687, 2019.

Eraker, B., Johannes, M., and Polson, N. G. The impact of jumps in volatility and returns. *Journal of Finance*, 58 (3):1269–1300, 2003.

Gneiting, T. and Raftery, A. E. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007. doi: 10.1198/016214506000001437.

Gordon, N. J., Salmond, D. J., and Smith, A. F. M. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings F (Radar and Signal Processing)*, 140(2):107–113, 1993.

Gu, A. and Dao, T. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv:2312.00752*, 2023.

Kim, S., Shephard, N., and Chib, S. Stochastic volatility: Likelihood inference and comparison with ARCH models. *Review of Economic Studies*, 65(3):361–393, 1998.

Merton, R. C. Option pricing when underlying stock returns are discontinuous. *Journal of Financial Economics*, 3 (1-2):125–144, 1976.

Parzen, E. Quantile probability and statistical data modeling. *Statistical Science*, 19(4):652–662, 2004.