Bridge Matching for Paired Tasks

Kirill Korolev*, Ian Maksimov*

1 Introduction

In this project a methodology, called Bridge Matching, for one-to-many generation is considered for solving an inpainting task. We compare three setups: 1) a generalized Brownian Bridge with a triangular schedule for a diffusion coefficient $g(t)$ 2) a Brownian Bridge with constant $g(t) = \sqrt{\gamma}$ 3) Flow Matching with $g(t) = 0$. Code implementation of the methods and a training pipeline can be found at **Github repository**.

2 Methods

We aim to construct a model that transports a transformed sample X_0 from the distribution p_0 to the original sample X_1 from p_1. At training, we receive data coupling (X_0, X_1) by sampling $X_1 \sim p_1 = p_{data}$ from the dataset and transforming it to $X_0 \sim p(\cdot|X_1)$. Then we construct an interpolation process X_t by sampling from $p_{X_t|X_0,X_1}(\cdot|X_0, X_1)$. In Bridge Matching conditional dynamics is defined by SDE, unlike Flow Matching, with the velocity field $f_t(x|x_0, x_1)$ and diffusion coefficient $g(t)$. In particular, we consider generalized Brownian Bridge — a pinned process, which is derived by applying Doob's h-transform to VE-SDE.

$$dX_t^{0,1} = \frac{g^2(t)(x_1 - X_t^{0,1})}{\int_t^1 g^2(s)ds}dt + g(t)dW_t \tag{1}$$

From the the literature we know that unconditional dynamics p_{X_t} is also generated by SDE with the vector field

$$f_t^*(x) = \mathbb{E}\left[f_t(X_t|X_0, X_1)\middle|X_t = x\right]$$

and the diffusion coefficient $g(t)$. Therefore, we can learn this conditional expectation by optimizing L^2 norm

$$\int_0^1 \left\|f_t^\theta(X_t) - \frac{g^2(t)(X_1 - X_t)}{\int_t^1 g^2(s)ds}\right\|_2^2 dt \tag{2}$$

where X_t is sampled from the conditional distribution $p_{X_t|X_0,X_1}(\cdot|X_0, X_1)$. Afterwards, inference is done by solving the following SDE using Euler-Maruyama discretization scheme

$$\begin{cases} dY_t = f_t^\theta(Y_t)dt + g(t)dW_t \\ Y_0 \sim p_0 \end{cases} \tag{3}$$

*These authors contributed equally to this work

In this work we consider three versions of a Brownian Bridge: 1) generalized Brownian Bridge with a triangular schedule of $g^2(t)$ 2) Brownian Bridge with a constant diffusion coefficient $g(t) = \sqrt{\gamma}$ 3) Flow Matching by setting $g(t) = 0$. In section 5.1 we derive a conditional distribution for a general case

$$p_{X_t|X_0,X_1}(x|x_0,x_1) = \mathcal{N}\left(x \middle| \frac{\bar{\sigma}_t^2}{\sigma_t^2+\bar{\sigma}_t^2}x_0 + \frac{\sigma_t^2}{\sigma_t^2+\bar{\sigma}_t^2}x_1, \frac{\sigma_t^2\bar{\sigma}_t^2}{\sigma_t^2+\bar{\sigma}_t^2}I\right) \tag{4}$$

where $\sigma_t^2 = \int_0^t g^2(s)ds$ and $\bar{\sigma}_t^2 = \int_t^1 g^2(s)ds$. Note that for the second case $g(t) = \sqrt{\gamma}$ it is simplified to

$$p_{X_t|X_0,X_1}(x|x_0,x_1) = \mathcal{N}\left(x \middle| (1-t)x_0 + tx_1, \gamma(1-t)tI\right) \tag{5}$$

We considered two model parametrizations: 1) prediction of a vector field 2) prediction of X_1 by utilizing that if we have optimal $D_t^*(x) = \mathbb{E}\left[X_1 \middle| X_t = x\right]$, then we can derive the vector field as

$$f_t^*(x) = \frac{g^2(t)\Big(D_t^*(x) - x\Big)}{\bar{\sigma}_t^2} \tag{6}$$

The training procedure is described in **Algorithm 1** below. Further in section 3 we discuss the sampling strategy for a timestep t and the choice of parametrization between f_t^θ and D_t^θ. As we mentioned earlier, sampling is done by solving SDE using Euler scheme and is expressed in **Algorithm 2**.

Algorithm 1 Training

Sample $X_1 \sim p_1$ and get a transformed sample $X_0 = Transform(X_1)$
Sample $t \sim sigmoid(\mathcal{N}(\mu, 1))$ or $U[0,1]$
Sample $X_t \sim p_{X_t|X_0,X_1}(\cdot|X_0, X_1)$
Backprop through $\nabla_\theta \|D_t^\theta(X_t) - X_1\|_2^2$ or $\nabla_\theta \|f_t^\theta(X_t) - \frac{g^2(t)(X_1 - X_t)}{\bar{\sigma}_t^2}\|_2^2$

Algorithm 2 Sampling

Receive X_0
$X_1 = SDESolve(X_0, 0 \to 1, f_t^\theta(\cdot), g(\cdot))$
return X_1

3 Experiments

We conducted experiments on FFHQ 64x64 dataset, which has 60k train and 10k test samples. During our work we considered an inpainting task, where we mask out top-half of the image. For the model we utilized SongUNet from EDM repository that has 60 million parameters with default configuration and trained it for 80k iterations, which took ~ 5 hours. We used Adam optimizer with learning rate warmup from 0 to 10^{-4} for 10k iterations and an exponential decay afterwards.

We chose to optimize D_t^θ mostly because it is easier to track model progress on predicted outputs and it resembles "denoiser" from a classical diffusion, while outputs from f_t^θ is less interpretable. Practically, we noticed in our experiments that training of the model predicting X_1 is more stable and samples were of better quality. Nevertheless, we faced a problem of overfitting, in particular, starting from some values of t sample X_1 is more likely to leak into interpolation X_t, making a prediction effortless to the model, which achieves almost zero loss, but with the poor sampling.

Figure 1: Leakage of original sample to the interpolated one. Model learns almost the identity transform.

Unlike unconditional generation, where we consider that noise is independent from the data sample and our task is easy at the beginning and at the end of sampling, here learning $\mathbb{E}[X_1|X_t]$ is gradually less complex with the increase of t. Therefore, instead of sampling timesteps uniformly during training, to encourage more initial timesteps we sample $t \sim sigmoid(\mathcal{N}(\mu, 1))$, where $\mu \in [-2.5, -1]$. In addition, we gradually increase learning rate at the beginning and use weight decay to stabilize training and reduce overfitting.

For a Brownian Bridge we used $\gamma = 0.1$, because in our experiments greater values led to quality degradation of samples. For a generalized Brownian Bridge we used a triangular schedule for a diffusion coefficient $g^2(t) = 2\beta_{max} \min(1-t, t)$, like in I^2SB paper. We select such β_{max} that matches maximum values of the variance in the interpolation process for two setups and set $\beta_{max} = 0.2$.

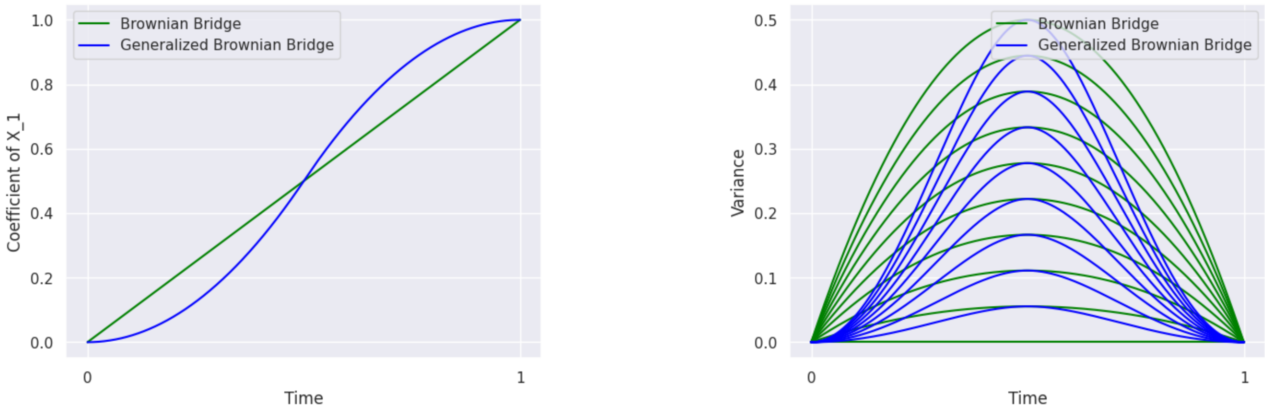

Figure 2: At the left coefficients of X_1 in the interpolation and at the right variance for different values of γ and β_{max} for Brownian Bridge and generalized Brownian Bridge respectively.

Apparently, because of our inductive bias that we want to learn model on timesteps near $t \approx 0$, we cannot exploit uniform grid during sampling. Therefore, we used EDM-like schedule with $\rho = 10, \sigma_{min} = 10^{-2}, \sigma_{max} = 1$ and 20 steps. An example of generation from a basic Brownian Bridge is depicted in Figure 4. More qualitative results can be found in the Appendix in Section 5.2. We calculated **FID** metric on a validation part of FFHQ and got values $30 - 40$ for BB and GBB models. Unfortunately, getting competitive values requires a great amount of computational time.

Figure 3: Samples from a trained model, which utilizes Brownian Bridge with a constant diffusion coefficient $g(t)$.

4 Conclusion

Eventually, Bridge Matching is rather suitable for solving paired tasks, when we desire one-to-many generation and defines easy to implement training and sampling procedures, but requires some computational resources for complex datasets like FFHQ and a subtle tuning of hyperparameters. Practically, we noticed that Bridge Matching converges faster than Flow Matching and regularizes model better by injecting noise.

Figure 4: Visual comparison between samples of three models: simple Brownian Bridge, generalized Brownian Bridge and Flow Matching.

5 Appendix

5.1 Derivation of conditional distribution for generalized Brownian Bridge

The SDE for the generalized Brownian bridge is as follows:

$$dX_t^{01} = \frac{g^2(t)(x_1 - X_t^{01})}{\int_t^1 g^2(s)ds}dt + g(t)dW_t \tag{7}$$

The process is Gaussian. Let us find its expectation and covariance matrix using discretization:

$$X_{t+h} = X_t + h\frac{g^2(t)(x_1 - X_t)}{\int_t^1 g^2(s)ds} + g(t)\sqrt{h}\epsilon \tag{8}$$

Taking the expectation of both sides and denoting $m_t := \mathbb{E}X_t$, we get:

$$m_{t+h} = m_t + h\frac{g^2(t)(x_1 - m_t)}{\int_t^1 g^2(s)ds} \tag{9}$$

$$dm_t = \frac{g^2(t)(x_1 - m_t)}{\int_t^1 g^2(s)ds}dt \tag{10}$$

$$\frac{dm_t}{dt} + \frac{g^2(t)}{\int_t^1 g^2(s)ds}m_t = \frac{g^2(t)}{\int_t^1 g^2(s)ds}x_1 \tag{11}$$

Let us define $\sigma_t^2 = \int_0^t g^2(s)ds$ and $\bar{\sigma}_t^2 = \int_t^1 g^2(s)ds$. Note that $\frac{d}{dt}\bar{\sigma}_t^2 = -g^2(t)$. Then the integrating factor is given by:

$$\exp\left(\int_0^t \frac{g^2(s)}{\int_s^1 g^2(u)du}ds\right) = \exp\left(\int_0^t \frac{-\frac{d}{ds}\bar{\sigma}_s^2}{\bar{\sigma}_s^2}ds\right) = \exp\left(-\log\bar{\sigma}_t^2\right) = \frac{1}{\bar{\sigma}_t^2} \tag{12}$$

Multiplying (11) by this factor and simplifying the left-hand side as a derivative of product, we obtain:

$$\frac{d}{dt}\left(\frac{m_t}{\bar{\sigma}_t^2}\right) = \frac{g^2(t)}{(\bar{\sigma}_t^2)^2}x_1 \tag{13}$$

$$m_t = \bar{\sigma}_t^2\left(x_1\int_0^t \frac{g^2(s)}{(\bar{\sigma}_s^2)^2}ds + C\right) \tag{14}$$

Considering the coefficient of x_1, we observe:

$$\bar{\sigma}_t^2\int_0^t \frac{g^2(s)}{(\bar{\sigma}_s^2)^2}ds = \bar{\sigma}_t^2\int_0^t \frac{-\frac{d}{ds}\bar{\sigma}_s^2}{(\bar{\sigma}_s^2)^2}ds = \bar{\sigma}_t^2\left(\frac{1}{\bar{\sigma}_t^2} - \frac{1}{\bar{\sigma}_0^2}\right) = \frac{\sigma_t^2}{\sigma_t^2 + \bar{\sigma}_t^2} \tag{15}$$

Finding C from the initial condition $m_0 = x_0$, we get $C = \frac{x_0}{\sigma_t^2 + \bar{\sigma}_t^2}$. Thus, we have:

$$m_t = \mathbb{E}X_t = \frac{\bar{\sigma}_t^2}{\sigma_t^2 + \bar{\sigma}_t^2} x_0 + \frac{\sigma_t^2}{\sigma_t^2 + \bar{\sigma}_t^2} x_1 \tag{16}$$

For the variance, we proceed analogously. Rewrite the discretization as follows:

$$X_{t+h} = X_t \left(1 - h \frac{g^2(t)}{\int_t^1 g^2(s) ds} \right) + h \frac{g^2(t)}{\int_t^1 g^2(s) ds} x_1 + g(t)\sqrt{h}\epsilon \tag{17}$$

Taking the variance of both sides and denoting $\mathbb{D}X_t := v_t$, we get the following ODE in the limit:

$$v_{t+h} = v_t \left(1 - 2h \frac{g^2(t)}{\int_t^1 g^2(s) ds} \right) + g^2(t)h + o(h) \tag{18}$$

$$dv_t = \left(\frac{-2g^2(t)}{\bar{\sigma}_t^2} v_t + g^2(t) \right) dt \tag{19}$$

Using the integrating factor $\frac{1}{(\bar{\sigma}_t^2)^2}$, which is derived analogously as for expectation ODE, and multiplying through, we get:

$$\frac{d}{dt} \left(\frac{v_t}{(\bar{\sigma}_t^2)^2} \right) = \frac{g^2(t)}{(\bar{\sigma}_t^2)^2} \tag{20}$$

Integrating both sides and noting that the integration constant is zero due to the initial condition $v_0 = 0$, we find the solution:

$$v_t = \frac{\sigma_t^2 \bar{\sigma}_t^2}{\sigma_t^2 + \bar{\sigma}_t^2} \tag{21}$$

Thus, the conditional distribution is given by:

$$p_{X_t|X_0,X_1}(x|x_0, x_1) = \mathcal{N}\left(x \middle| \frac{\bar{\sigma}_t^2}{\sigma_t^2 + \bar{\sigma}_t^2} x_0 + \frac{\sigma_t^2}{\sigma_t^2 + \bar{\sigma}_t^2} x_1, \frac{\sigma_t^2 \bar{\sigma}_t^2}{\sigma_t^2 + \bar{\sigma}_t^2} I \right) \tag{22}$$

5.2 Qualitative Results


Original
Transformed
Predicted


Original
Transformed
Predicted


Original
Transformed
Predicted