

786 Contents

787	A Summary of Notation	20
788	B Proofs	21
789	B.1 Proof of Theorem 3.1	21
790	B.2 Proof of Proposition 3.2	23
791	B.3 Proof of Proposition 3.4	26
792	B.4 Proof of Theorem 3.5	27
793	C Gaussian Path	29
794	D Extended Related Works	31
795	E Additional Details on Experiments	32
796	E.1 General Information	32
797	E.2 Lotka-Volterra	32
798	E.3 Gulf of Mexico	33
799	E.4 Beijing air quality	33
800	E.5 Single sequencing	34
801	E.6 Ablation study on σ	34

802 A Summary of Notation

803 In the Table below, we summarize the notation used throughout our work.

Table 6: Notation

t	Time coordinate
x_t	Position
v_t	Velocity
a_t	Acceleration
m_t	Augmented Variable
$q_n(x_n)$	n^{th} Positional marginal
$\xi_n(v_n)$	n^{th} Velocity marginal
$\pi_n(x_n, v_n)$	n^{th} Augmented marginal
$p_t(x_t, v_t)$	Augmented probability path
$\lambda^{(n)}$	Coefficients of impact of future pinned points of the n^{th} segment
$C^{(n)}(t)$	Functions shaping the cond. bridge of the n^{th} segment
$\{x_n\}$	Coupling over n marginals
$V_t(m_t)$	Value Function
P_t, r_t	Value function second and first order approximations
μ_t	Gaussian path mean vector
Σ_t	Gaussian path covariance matrix
L_t	Cholesky decomposition on the cov. matrix

804 B Proofs

805 B.1 Proof of Theorem 3.1

806 **Theorem B.1** (SOC representation of Multi-Marginal Momentum Brownian Bridge (3MBB)). *Con-*
807 *sider the following momentum system interpolating among multiple marginals*

$$u_t^* = \begin{bmatrix} 0 \\ a_t^* \end{bmatrix} \in \arg \min_{a_t} \int_0^1 \frac{1}{2} \|a_t\|^2 dt + \sum_{n=1}^N (m_n - \bar{m}_n)^\top R (m_n - \bar{m}_n) \quad (11)$$

$$s.t \quad dm_t = A m_t dt + u_t + g dW_t, \quad m_0 = \bar{m}_0. \quad (12)$$

808 We define the value function as $V_t(m_t) := \frac{1}{2} m_t^\top P_t^{-1} m_t + m_t^\top P_t^{-1} r_t$, where P_t, r_t are the second-
809 and first-order approximations, respectively. This formulation admits the following optimal control
810 expression $u_t^*(m_t) = -g g^\top P_t^{-1} (m_t + r_t)$. For the multi-marginal bridge with $\{\bar{m}_n\}$ fixed at $\{t_n\}$,
811 for $n \in \{0, 1, \dots, N\}$, the dynamics of P_t and r_t obey the following backward ODEs

$$\dot{P}_t = A P_t + P_t A^\top - \sigma_t \sigma_t^\top, \quad P_n = (P_{n+}^{-1} + R)^{-1}, \text{ with } P_{N+} = 0 \quad (13)$$

$$\dot{r}_t = -A r_t, \quad r_n = P_n (P_{n+}^{-1} r_{n+} - R \bar{m}_n) \quad (14)$$

812 where $P_{n+} := \lim_{t \rightarrow t_n^+} P_t$, and $r_{n+} := \lim_{t \rightarrow t_n^+} r_t$, for $t \in \{s : s \in (t_1, t_2) \cup (t_2, t_3), \dots\}$.

813 *Proof.* We start our analysis by considering the second-order approximation of the value function:

$$V(t, m_t) = \frac{1}{2} m_t^\top Q_t m_t + r_t^\top Q_t m_t \quad (15)$$

814 where Q_t and r_t serve as second and first-order approximations. From the Bellman principle and
815 application of the Ito's Lemma to the value function, we obtain the Hamilton-Jacobi-Bellman (HJB)
816 Equation:

$$V_t + \min_u \left[\frac{1}{2} \mathbb{E} [\|u_t\|^2 dt] + V_m^\top (A m_t + u_t) \right] + \frac{1}{2} \text{Tr}(V_{mm} g g^\top) = 0 \quad (16)$$

817 Solving for the optimal control u_t^* , we obtain:

$$u_t^* = -g g^\top V_m = -g g^\top Q (m_t + r_t) \quad (17)$$

818 Thus plugging it back to the HJB, we can rewrite it

$$V_t - \frac{1}{2} V_{mm} g g^\top V_m + V_m^\top A m + \frac{1}{2} \text{Tr}(V_{mm} g g^\top) = 0 \quad (18)$$

Recall the definition for the value function

$$V(t, m_t) = \frac{1}{2} m_t^\top Q m_t + r_t^\top Q m_t$$

819 Substituting for the definition of the value function in the HJB yields the following PDE

$$\frac{1}{2} m_t^\top \dot{Q} m_t + \dot{r}_t^\top Q m_t + r_t^\top \dot{Q} m_t - \frac{1}{2} m_t^\top Q g g^\top Q m_t - r_t^\top Q g g^\top Q m_t + m_t^\top Q A m_t + r_t^\top Q A m_t + \frac{1}{2} \text{Tr}(V_{mm} g g^\top) = 0 \quad (19)$$

820 Grouping the terms of the PDE quadratic in m_t , we obtain the Riccati Equation for Q_t

$$-\dot{Q} = A^\top Q + Q A - Q g g^\top Q \quad (20)$$

821 Then, grouping the linear terms yields

$$\dot{r}_t^\top Q + r_t^\top \dot{Q} - r_t^\top Q g g^\top Q + r_t^\top Q A = 0 \quad (21)$$

822 Now, notice that substituting the Riccati in Eq. 20 into Eq. 21, one obtains

$$\dot{r}_t = -A r_t \quad (22)$$

823 The solution of this ODE is

$$r_t = \Phi(t, s) r_s \quad (23)$$

where $\Phi(t, s)$ is the state transition function of the following dynamics $dm_t = A(t)m_t dt$, from t to s and is defined as $\Phi(t, s) = \begin{bmatrix} 1 & t-s \\ 0 & 1 \end{bmatrix}$. Finally, we define $P(t) := Q(t)^{-1}$, and modify the Ricatti in Eq. 20 as follows

$$\begin{aligned} -\dot{Q} &= A^\top Q + Q^{-1}QAQ^{-1} - Q\mathbf{g}\mathbf{g}^\top Q \\ -Q^{-1}\dot{Q}Q^{-1} &= Q^{-1}A^\top QQ^{-1} + Q^{-1}QAQ^{-1} - Q^{-1}Q\mathbf{g}\mathbf{g}^\top QQ^{-1} \\ \dot{P} &= AP + PA^\top - \mathbf{g}\mathbf{g}^\top \end{aligned} \quad (24)$$

yielding the Lyapunov equation

$$\dot{P} = AP + PA^\top - \mathbf{g}\mathbf{g}^\top \quad (25)$$

Therefore, we have proved the desired ODEs for the first and second-order approximations r_t, P_t of the value function. Now, we have to determine the expressions for the terminal conditions for the ODEs in each segment.

Note that it follows from the dynamic principle that these ODEs are backward, therefore ensuing segments will affect previous. These terminal conditions carry the information from the subsequent segments. More specifically, assume a multi-marginal process with $\{\bar{m}_{n+1}\}$ pinned at $\{t_{n+1}\}$. Now let us consider the two segments from both sides:

- Segment n : for $t \in [t_n, t_{n+1}]$
- Segment $n+1$: for $t \in [t_{n+1}, t_{n+2}]$

To solve P_t, r_t for Segment n , we have to account for the effect of Segment $n+1$. To compute the value function at t_{n+1} , accounting for the impact from Segment $n+1$

$$V_{n+1} = V_{n+1+} + \frac{1}{2}(m_{n+1} - \bar{m}_{n+1})^T R_{n+1}(m_{n+1} - \bar{m}_{n+1}) \quad (26)$$

where the first term encapsulates the impact from Segment $n+1$. More explicitly, for the Segment $n+1$, at $t = t_{n+1}$, we define the value function as

$$\begin{aligned} V_{n+1+} &= \frac{1}{2}m_{n+1+}^T Q_{n+1+} m_{n+1+} + r_{n+1+} Q_{n+1+} m_{n+2+} \\ &= \frac{1}{2}m_{n+1+}^T Q_{n+1+} m_{n+1+} + \Phi(n+1, n+2)\bar{m}_{n+2} Q_{n+1+} m_{n+2+} \end{aligned} \quad (27)$$

Therefore, for Segment n the value function at the terminal time t_{n+1} is given by

$$\begin{aligned} V_{n+1} &= V_{n+1+} + \frac{1}{2}(m_{n+1} - \bar{m}_{n+1})^T R_{n+1}(m_{n+1} - \bar{m}_{n+1}) \\ &= \frac{1}{2}m_{n+1}^\top (Q_{n+1+} + R_{n+1})m_{n+1} + (\Phi(n+1, T)\bar{m}_T Q_{n+1+} \\ &\quad - \bar{m}_{n+1} R_{n+1})m_{n+1} + \text{const.} \\ &= \frac{1}{2}m_{n+1}^T Q_{n+1} m_{n+1} + r_{n+1} Q_{n+1} m_{n+1} + \text{const.} \end{aligned} \quad (28)$$

This suggests that the terminal constraints for the Ricatti equation Q_t , and the reference dynamics vector r_t are given by

$$Q_{n+1} := Q_{n+1+} + R_{n+1} \quad (29)$$

$$r_{n+1} := (\Phi(t_{n+1}, T)\bar{m}_T Q_{n+1+} - \bar{m}_{n+1} R_{n+1})Q_{n+1}^{-1} \quad (30)$$

Lastly, recall that the Lyapunov function is defined as $P_t = Q_t^{-1}$, thus the terminal constraints with respect to the Lyapunov function are given by

$$P_{n+1} = (P_{n+1+}^{-1} + R_{n+1})^{-1}, \quad r_{n+1} = P_{n+1}(P_{n+1+}^{-1} r_{n+1+} - R_{n+1}\bar{m}_{n+1}) \quad (31)$$

This implies that r_t —and hence the optimal control u_t^* —would depend on all preceding pinned points after t $\{\bar{m}_j : t_j \geq t\}$. Finally, notice that for the last segment $t \in [t_{N-1}, t_N]$, it holds that $P_{N+} = 0$, since there is no effect from any subsequent segment, hence $P_{t_N} = R_{t_N}^{-1}$, and r_{t_N} simplifies to $-m_{t_N}$. \square

850 B.2 Proof of Proposition 3.2

851 **Proposition B.2.** Let $R = \begin{bmatrix} \frac{1}{c} & 0 \\ 0 & c \end{bmatrix}$. At the limit when $c \rightarrow 0$, the conditional acceleration in Eq. 3
852 admits an analytic form:

$$a_t^*(m_t | \{\bar{x}_{n+1} : t_{n+1} \geq t\}) = C_1^n(t)(x_t - \bar{x}_{n+1}) + C_2^n(t)v_t + C_3^n(t) \sum_{j=n+1}^N \lambda_j \bar{x}_j, \quad t \in [t_n, t_{n+1}) \quad (32)$$

853 where $\{\bar{x}_{n+1} : t_{n+1} \geq t\}$ signifies the bridge is conditioned on the set of the ensuing points, λ_j are
854 static coefficients and $C_1^n(t), C_2^n(t), C_3^n(t)$ are time-varying coefficients specific to each segment.

855 *Proof.* We start our analysis with the last segment $N - 1$, and then move to derive the formulation
856 for an arbitrary preceding segment n .

857 **Segment $N - 1$:** $t \in [t_{N-1}, T]$

858 For the last segment, the terminal constraint $t \in [t_{N-1}, T]$ is given by $P_N = R_N^{-1}$, and hence
859 r_N simplifies to $-m_N$. Solving the backward differential equation P_t , for $t \in [t_{N-1}, T]$, with
860 $P_T = P_N = R_N^{-1}$, yields

$$P(t) = \begin{bmatrix} -\frac{\sigma^2}{2}(t-T)^3 + \frac{(t-T)^2}{c} + c & -\frac{\sigma^2}{2}(t-T)^2 + \frac{(t-T)}{c} \\ -\frac{\sigma^2}{2}(t-T)^2 + \frac{(t-T)}{c} & -\sigma^2(t-T) + \frac{1}{c} \end{bmatrix} \quad (33)$$

861 Additionally, solving for r_t for $t \in [t_{N-1}, T]$, with $r_T = -\bar{m}_T$ yields:

$$r_t = \Phi(t, T)\bar{m}_T \quad (34)$$

862 where $\Phi(t, s)$ is the transition matrix of the following dynamics $dm_t = A(t)m_t dt$, and is defined as

863 $\Phi(t, s) = \begin{bmatrix} 1 & t-s \\ 0 & 1 \end{bmatrix}$. Plugging Eq. 33, and 34 into Eq. 17 yields

$$u_t^* = \begin{bmatrix} 0 \\ \frac{3}{(t-T)^2}(\bar{x}_T - x_t) - \frac{3}{T-t}v_t \end{bmatrix}, \quad \forall t \in [t_{N-1}, T] \quad (35)$$

864 Therefore, regardless of the total number of marginals the C -functions for the last segments are
865 always: $C_1^{(N-1)}(t) = -\frac{3}{(T-t)^2}$, $C_2^{(N-1)}(t) = \frac{3}{T-t}$, $C_3^{(N-1)}(t) = 0$.

866 **Segment n :** $t \in [t_n, t_{n+1}]$

867 Now, we move to derive the conditional acceleration, for an arbitrary segment n , with $n < N - 1$ i.e.
868 n is not the last segment. Let us recall the optimal control formulation from Eq. (17)

$$u_t^*(m_t) = -\mathbf{g}\mathbf{g}^\top P_t^{-1}(m_t + \Phi(t, t_{n+1})r_{n+1}) \quad (36)$$

869 For convenience, let us define the following functions corresponding to the n^{th} segment: $t \in [t_n, t_{n+1}]$

$$\begin{aligned} z_1^{(n)}(t) &= 3t - 3t_{n+1} - 3 & z_2^{(n)}(t) &= 3t - 4 - 3t_{n+1} \\ z_3^{(n)}(t) &= 6t - 6t_{n+1} - 3 & z_4^{(n)}(t) &= 6t - 6t_{n+1} - 4 \\ z_5^{(n)}(t) &= 4t - 4t_{n+1} - 3 & z_6^{(n)}(t) &= 4t - 4 - 4t_{n+1} \\ z_7^{(n)}(t) &= 6t - 6t_{n+1} + 3 & z_8^{(n)}(t) &= 6t - 6t_{n+1} + 4 \end{aligned} \quad (37)$$

870 For this arbitrary segment n , we solve the backward ODE P_t , with the corresponding terminal, using
871 an ODE solution software.

$$\dot{P}_t = AP_t + P_t A^T - \sigma_t \sigma_t^T, \quad P_n = (P_{n+}^{-1} + R)^{-1}, \quad \text{with } P_{N+} = 0 \quad (38)$$

872 Given the structure of $R = \begin{bmatrix} \frac{1}{c} & 0 \\ 0 & c \end{bmatrix}$, with $c \rightarrow 0$ this leads us to the following expression for P_t , for
873 $t \in [t_n, t_{n+1})$

$$P_t = \begin{bmatrix} \sigma^2 \frac{(t-t_{n+1})^3 (\alpha^{(n)} z_1^{(n)}(t) + \beta^{(n)} z_2^{(n)}(t))}{2(\alpha^{(n)} z_5^{(n)}(t) + \beta^{(n)} z_6^{(n)}(t))} & \sigma^2 \frac{(t-t_{n+1})^2 (\alpha^{(n)} z_1^{(n)}(t) + \beta^{(n)} z_2^{(n)}(t))}{2(\alpha^{(n)} z_3^{(n)}(t) + \beta^{(n)} z_4^{(n)}(t))} \\ \sigma^2 \frac{(t-t_{n+1})^2 (\alpha^{(n)} z_1^{(n)}(t) + \beta^{(n)} z_2^{(n)}(t))}{2(\alpha^{(n)} z_3^{(n)}(t) + \beta^{(n)} z_4^{(n)}(t))} & 2\sigma^2 \frac{(t-t_{n+1}) (\alpha^{(n)} z_1^{(n)}(t) + \beta^{(n)} z_2^{(n)}(t))}{(\alpha^{(n)} z_7^{(n)}(t) + \beta^{(n)} z_8^{(n)}(t))} \end{bmatrix} \quad (39)$$

874 where $\alpha^{(n)}, \beta^{(n)}$ are segment-specific coefficients that shape the conditional bridge for the given
 875 segment n . These are recursively computed using the coefficients of the subsequent segment $n + 1$,
 876 as follows:

$$\begin{aligned}\alpha^{(n)} &= \alpha^{(n+1)} z_1^{(n)}(t_{n+1}) + \beta^{(n+1)} z_2^{(n)}(t_{n+1}) \\ \beta^{(n)} &= 4(\alpha^{(n+1)} + \beta^{(n+1)})\end{aligned}\quad (40)$$

877 From the expression of P_t , we can obtain its inverse

$$P_t^{-1} = \frac{1}{\sigma^2} \begin{bmatrix} \frac{6(\alpha^{(n)} z_7^{(n)}(t) + \beta^{(n)} z_8^{(n)}(t))}{(t-t_{n+1})^3 * (\alpha^{(n)} z_1^{(n)}(t) + \beta^{(n)} z_2^{(n)}(t))} & \frac{-3(\alpha^{(n)} z_3^{(n)}(t) + \beta^{(n)} z_4^{(n)}(t))}{(t-t_{n+1})^2 * (\alpha^{(n)} z_1^{(n)}(t) + \beta^{(n)} z_2^{(n)}(t))} \\ \frac{-3(\alpha^{(n)} z_3^{(n)}(t) + \beta^{(n)} z_4^{(n)}(t))}{(t-t_{n+1})^2 * (\alpha^{(n)} z_1^{(n)}(t) + \beta^{(n)} z_2^{(n)}(t))} & \frac{3(\alpha^{(n)} z_5^{(n)}(t) + \beta^{(n)} z_6^{(n)}(t))}{(t-t_{n+1}) * (\alpha^{(n)} z_1^{(n)}(t) + \beta^{(n)} z_2^{(n)}(t))} \end{bmatrix} \quad (41)$$

878 Hence, we can compute the terms: i) $\mathbf{g}\mathbf{g}^\top P_t^{-1}$, ii) $\mathbf{g}\mathbf{g}^\top P_t^{-1} \Phi(t, t_{n+1})$ in the optimal control formula-
 879 tion as follows:

$$\mathbf{g}\mathbf{g}^\top P_t^{-1} = \begin{bmatrix} 0 & 0 \\ \frac{-3(\alpha^{(n)} z_3^{(n)}(t) + \beta^{(n)} z_4^{(n)}(t))}{(t-t_{n+1})^2 * (\alpha^{(n)} z_1^{(n)}(t) + \beta^{(n)} z_2^{(n)}(t))} & \frac{3(\alpha^{(n)} z_5^{(n)}(t) + \beta^{(n)} z_6^{(n)}(t))}{(t-t_{n+1}) * (\alpha^{(n)} z_1^{(n)}(t) + \beta^{(n)} z_2^{(n)}(t))} \end{bmatrix} \quad (42)$$

$$\mathbf{g}\mathbf{g}^\top P_t^{-1} \Phi(t, t_{n+1}) = \begin{bmatrix} 0 & 0 \\ \frac{-3(\alpha^{(n)} z_3^{(n)}(t) + \beta^{(n)} z_4^{(n)}(t))}{(t-t_{n+1})^2 * (\alpha^{(n)} z_1^{(n)}(t) + \beta^{(n)} z_2^{(n)}(t))} & \frac{6(\alpha^{(n)} + \beta^{(n)})}{(\alpha^{(n)} z_1^{(n)}(t) + \beta^{(n)} z_2^{(n)}(t))} \end{bmatrix} \quad (43)$$

880 Therefore, we can rewrite the optimal control in Eq. (17) as

$$u_t^* = \begin{bmatrix} 0 & 0 \\ C_1^{(n)}(t) & C_2^{(n)}(t) \end{bmatrix} \begin{bmatrix} \mathbf{x}_t \\ \mathbf{v}_t \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ C_1^{(n)}(t) & C_3^{(n)}(t) \end{bmatrix} r_{n+1}, \forall t \in [t_n, t_{n+1}] \quad (44)$$

881 where we have that

$$\begin{aligned}C_1^{(n)}(t) &= \frac{-3(\alpha^{(n)} z_3^{(n)}(t) + \beta^{(n)} z_4^{(n)}(t))}{(t-t_{n+1})^2 * (\alpha^{(n)} z_1^{(n)}(t) + \beta^{(n)} z_2^{(n)}(t))} \\ C_2^{(n)}(t) &= \frac{3(\alpha^{(n)} z_5^{(n)}(t) + \beta^{(n)} z_6^{(n)}(t))}{(t-t_{n+1}) * (\alpha^{(n)} z_1^{(n)}(t) + \beta^{(n)} z_2^{(n)}(t))} \\ C_3^{(n)}(t) &= \frac{6(\alpha^{(n)} + \beta^{(n)})}{(\alpha^{(n)} z_1^{(n)}(t) + \beta^{(n)} z_2^{(n)}(t))}\end{aligned} \quad (45)$$

882 Now we proceed to the computation of the term r_{n+1} , for which it holds from Eq. (13), that

$$r_{n+1} = P_{n+1}(P_{n+1}^{-1} r_{n+1+} - R\bar{m}_{n+1}) \quad (46)$$

883 where the term $r_{n+1+} = \Phi(t_{n+1}, t_{n+2}) r_{n+2}$ carries the impact from the future segments. Therefore,
 884 the first term recursively introduces the impact of further future pinned points through

$$P_{n+1} P_{n+1+}^{-1} r_{n+1+} = P_{n+1} P_{n+1+}^{-1} \Phi(t_{n+1}, t_{n+2}) r_{n+2} \quad (47)$$

885 Since $P_{n+1} P_{n+1+}^{-1} \Phi(t_{n+1}, t_{n+2})$ is also independent of time, we have that

$$P_{n+1} P_{n+1+}^{-1} \Phi(t_{n+1}, t_{n+2}) = \begin{bmatrix} 0 & 0 \\ \lambda^{(n)} & 0 \end{bmatrix} \quad (48)$$

886 where λ is some static coefficient, specific for the n^{th} segment, i.e., different segments are character-
 887 ized by different λ coefficients.

888 The structure of this matrix in Eq. (48) suggests that r_t will be dependent only on the positional
 889 constraints, when multiplied with \bar{m}_j for $j = \{n+2, \dots, N\}$. Finally, given the linearity of the
 890 dynamics of r_t , we can recursively add the impact of more pinned points

$$P_{n+1} P_{n+1+}^{-1} \Phi(t_{n+1}, t_{n+2}) r_{n+2} = P_{n+1} P_{n+1+}^{-1} \Phi(t_{n+1}, t_{n+2}) (P_{n+2}(P_{n+2+}^{-1} r_{n+2+} - R\bar{m}_{n+2})) \quad (49)$$

891 This recursion leads to r_t being expressed as a linear combination of those future pinned points,
892 through the following expression

$$P_{n+1}P_{n+1}^{-1}r_{n+1} = \begin{bmatrix} 0 \\ \sum_{j=n+2}^N \lambda_j^{(n)} \bar{x}_j \end{bmatrix} \quad (50)$$

893 Subsequently, computation of P_{n+1} from $P_{n+1} = (P_{n+1} + R)^{-1}$ along with the diagonal structure
894 of R also leads to

$$P_{n+1} R \bar{m}_{n+1} = \begin{bmatrix} -x_{n+1} \\ \kappa^{(n)} x_{n+1} \end{bmatrix} \quad (51)$$

895 where $\kappa^{(n)}$ is also some static coefficient, specific to the n^{th} segment. Consequently, this leads to the
896 following expression for r_{n+1}

$$r_{n+1} = \begin{bmatrix} -\bar{x}_{n+1} \\ \kappa^{(n)} \bar{x}_{n+1} + \sum_{j=n+2}^N \lambda_j^{(n)} \bar{x}_j \end{bmatrix} \quad (52)$$

897 or equivalently

$$r_{n+1} = \begin{bmatrix} -\bar{x}_{n+1} \\ \sum_{j=n+1}^N \lambda_j^{(n)} \bar{x}_j \end{bmatrix} \quad (53)$$

898 where we defined $\kappa^{(n)} = \lambda_{n+1}^{(n)}$. This implies that r_t —and hence the optimal control u_t^* —depends
899 on all preceding pinned points after t $\{\bar{m}_{n+1} : t_{n+1} \geq t\}$, as a linear combination of these points. It
900 is found that the elements of the vector $\lambda^{(n)} = [\lambda_{n+1}, \lambda_{n+2}, \dots, \lambda_N]$ depend only on the number
901 of accounted pinned points \bar{x}_j , and decay exponentially as this number increases, as illustrated in
902 Figure 10. In other words, the values of $\lambda_j^{(n)}$ decrease the further the corresponding \bar{x}_j is located
903 from the segment whose bridge we compute.

904 *Remark B.3.* It is highlighted that the sole dependency of the coefficients $\alpha^{(n)}, \beta^{(n)}, \lambda^{(n)}$ is the
905 number of future marginals.

906 □

907 B.2.1 Examples of Multi-Marginal Bridges

908 At this point, we provide examples of multi-marginal conditional bridges, elucidating that the structure
909 of each segment is governed by the number of future marginals. For simplicity, let us denote with
910 $\alpha^{(n)}, \beta^{(n)}, \lambda^{(n)}$ the segment-specific coefficients corresponding to the n^{th} segment: $t \in [t_n, t_{n+1})$.

911 **2-marginal Bridge** The formulation for a two-marginal bridge coincides with the segment $N - 1$
912 for a multi-marginal bridge, when $N = 2$, and $T = 1$. More specifically, we have:

$$a^*(m_t | \bar{x}_1) = \frac{3}{(t-1)^2}(\bar{x}_T - x_t) - \frac{3}{1-t}v_t, \forall t \in [0, 1) \quad (54)$$

913 **3-marginal Bridge** The formulation of the 3-marginal bridge is given by

$$\begin{cases} a^*(m_t | \bar{x}_2) = \frac{3}{(T-t)^2}(\bar{x}_2 - x_t) - \frac{3}{T-t}v_t, & t \in [t_1, T) \\ a^*(m_t | \bar{x}_1, \bar{x}_2) = \frac{-18t - 18t_1 - 12}{(t-1)^2(3t - 3t_1 - 4)}(x_t - \bar{x}_1) + \frac{12t - 12t_1 - 12}{(t-1)(3t - 3t_1 - 4)}v_t \\ \quad + \frac{6}{(3t - 3t_1 - 4)}(\bar{x}_2 - \bar{x}_1), & t \in [t_0, t_1) \end{cases} \quad (55)$$

914 Notice that for $t_1 = 1$, and $t_2 = 2$, we derive the same expression as in the Example in Section 3.
915 Additionally, we see that the segment $t \in [t_0, t_1)$ is obtained by our generalized formula for $\alpha = 0$,
916 $\beta = 1$, and coincides with the expression of the $N - 2$ segment. Finally, it is verified that the last
917 segment shares the same formulation with the same coefficients as the bridge of the 2-marginal case.

918 **4-marginal Bridge** The 4-marginal bridge further illustrates how the structure of each segment
 919 depends on the number of future marginals. In particular, following Remark B.3, the last two segments
 920 $t \in [t_2, T)$ and $t \in [t_1, t_2)$ share the same formulation as in the 3-marginal bridge, since they are
 921 conditioned on 1 and 2-future marginals, respectively. To compute the first segment, $t \in [t_0, t_1)$, we
 922 find that $\alpha^{(0)} = 4$, $\beta^{(0)} = 4$, and the $\lambda^{(0)}$ vector to be: $\lambda^{(0)} = [-1.25, 1.5, -0.25]^\top$. This results
 923 in the following bridge formulation:

$$\left\{ \begin{array}{ll} a^*(m_t | \bar{x}_3) = \frac{3}{(T-2)^2}(\bar{x}_3 - x_t) - \frac{3}{T-t}v_t, & t \in [t_2, T) \\ a^*(m_t | \bar{x}_2, \bar{x}_3) = \frac{-18t - 18t_2 - 12}{(t-t_2)^2(3t-3t_2-4)}(x_t - \bar{x}_2) \\ \quad + \frac{12t - 12t_2 - 12}{(t-t_2)(3t-3t_2-4)}v_t + \frac{6}{(3t-3t_2-4)}(\bar{x}_3 - \bar{x}_2), & t \in [t_1, t_2) \\ a^*(m_t | \bar{x}_1, \bar{x}_2, \bar{x}_3) = \frac{-36t + 36t_1 - 21}{(t-t_1)^2(6t-6t_1-7)}(x_t - \bar{x}_1) + \frac{24t - 24t_1 - 21}{(t-t_1)(6t-6t_1-7)}v_t \\ \quad + \frac{12}{(6t-6t_1-7)}(-0.25\bar{x}_3 + 1.5\bar{x}_2 - 1.25\bar{x}_1) & t \in [t_0, t_1) \end{array} \right. \quad (56)$$

924 **5-marginal Bridge** It is easy to see that the last three segments of the 5-marginal bridge follow the
 925 same structure as in the 4-marginal case. For example, based on Remark B.3, the coefficients for the
 926 third-to-last segment, $t \in [t_1, t_2)$, are $\alpha_1 = \beta_1 = 4$, and the vectors $\lambda^{(2)}$ and $\lambda^{(1)}$, corresponding to
 927 the segments $t \in [t_2, t_3)$ and $t \in [t_1, t_2)$, respectively, match those of the third-to-last and second-to-
 928 last segments in the 4-marginal bridge. Finally, for the first segment, $t \in [0, t_1)$, we compute that:
 929 $\alpha^{(0)} = 28$, $\beta^{(0)} = 32$, and $\lambda^{(0)} = [-1.267, 1.6, -0.4, 0.067]^\top$. Substituting these coefficients into
 930 Eq. (45) yields the corresponding bridge formulation.

$$\left\{ \begin{array}{ll} a^*(m_t | \bar{x}_4) = \frac{3}{(T-t)^2}(\bar{x}_4 - x_t) - \frac{3}{T-t}v_t, & t \in [t_3, T) \\ a^*(m_t | \bar{x}_3, \bar{x}_4) = \frac{-18t - 18t_3 - 12}{(t-t_3)^2(3t-3t_3-4)}(x_t - \bar{x}_3) \\ \quad + \frac{12t - 12t_3 - 12}{(t-t_3)(3t-3t_3-4)}v_t + \frac{6}{3t-3t_3-4}(\bar{x}_4 - \bar{x}_3), & t \in [t_2, t_3) \\ a^*(m_t | \bar{x}_2, \bar{x}_3, \bar{x}_4) = \frac{-36t + 36t_2 - 21}{(t-t_2)^2(6t-6t_2-7)}(x_t - \bar{x}_2) + \frac{24t - 24t_2 - 21}{(t-t_2)(6t-6t_2-7)}v_t \\ \quad + \frac{12}{(6t-6t_1-7)}(-0.25\bar{x}_4 + 1.5\bar{x}_3 - 1.25\bar{x}_2), & t \in [t_1, t_2) \\ a^*(m_t | \bar{x}_1, \bar{x}_2, \bar{x}_3, \bar{x}_4) = -\frac{6(45t - 45t_1 - 26)}{(t-t_1)^2(45t-45t_1-52)}(x_t - \bar{x}_1) + \frac{12(15t - 15t_1 - 13)}{(t-t_1)(45t-45t_1-52)}v_t \\ \quad + \frac{90}{45t-45t_1-52}(-1.267\bar{x}_1 + 1.6\bar{x}_2 - 0.4\bar{x}_3 + 0.067\bar{x}_4), & t \in [t_0, t_1) \end{array} \right. \quad (57)$$

931 B.3 Proof of Proposition 3.4

932 **Proposition B.4.** Let us define the marginal path p_t as a mixture of bridges $p_t(m_t) =$
 933 $\int p_t|\{\bar{x}_n\}(m_t|\{\bar{x}_n : t_n \geq t\})dq(\{x_n\})$, where $p_t|\{\bar{x}_n\}(m_t|\{\bar{x}_n : t_n \geq t\})$ is the conditional proba-
 934 bility path associated with the solution of the 3MBB path in Eq 6. The parameterized acceleration

935 that satisfies the FPE prescribed by the p_t is given by

$$a_t(t, m_t) = \frac{\int a_t|\{\bar{x}_n\} p_t|\{\bar{x}_n\}(m_t|\{\bar{x}_n : t_n \geq t\}) dq(\{x_n\})}{p_t} \quad (58)$$

936 This suggests that the minimization of the variational gap to match a_t^θ given p_t is given by

$$\min_{\theta} \mathbb{E}_{q(\{x_n\})} \mathbb{E}_{p_t|\{\bar{x}_n\}} \left[\int_0^1 \|a_t|\{\bar{x}_n\} - a_t^\theta\|^2 dt \right] \quad (59)$$

937 *Proof.* We want to show that the acceleration from Eq. 9 preserves the prescribed path p_t . The
938 momentum Fokker Plank Equation (FPE) is given by

$$\partial_t p_t(m_t) = -v_t \nabla_x p_t(m_t) - \nabla_v (a_t(m_t) p_t(m_t)) + \frac{1}{2} \sigma^2 \Delta_v p_t(m_t) \quad (60)$$

939 We let the marginal be constructed as a mixture of conditional probability paths conditioned on a
940 collection of pinned points $\{\bar{x}_n\}_{n \in [0, N]}$, $p_t = \int p_t(m_t|\{\bar{x}_n\}) q(\{\bar{x}_n\}) dx_0 dx_1 \dots dx_N$. Using this
941 definition for the marginal path, one obtains that

$$\begin{aligned} \partial_t p_t(m_t) &= \partial_t \int q(\{\bar{x}_n\}) p_t(m_t|\{\bar{x}_n\}) d\{\bar{x}_n\} = \int q(\{\bar{x}_n\}) \partial_t p_t(m_t|\{\bar{x}_n\}) d\{\bar{x}_n\} \\ v_t \nabla_x p_t(m_t) &= v_t \nabla_x \int q(\{\bar{x}_n\}) p_t(m_t|\{\bar{x}_n\}) d\{\bar{x}_n\} = \int q(\{\bar{x}_n\}) [v_t \nabla_x p_t(m_t|\{\bar{x}_n\})] d\{\bar{x}_n\} \\ \Delta_v p_t(m_t) &= \nabla_v \cdot (\nabla_v p_t(m_t)) = \nabla_v \cdot \left(\nabla_v \int q(\{\bar{x}_n\}) p_t(m_t|\{\bar{x}_n\}) d\{\bar{x}_n\} \right) \\ &= \int q(\{\bar{x}_n\}) \left[\nabla_v \cdot (\nabla_v \int p_t(m_t|\{\bar{x}_n\}) d\{\bar{x}_n\}) \right] d\{\bar{x}_n\} \\ &= \int q(\{\bar{x}_n\}) \Delta_v p_t(m_t|\{\bar{x}_n\}) d\{\bar{x}_n\} \end{aligned} \quad (61)$$

942 Hence it remains to be checked whether the following equality holds

$$a_t(m_t) p_t(m_t) = \int q(\{\bar{x}_n\}) [a_t|1(m_t|\{\bar{x}_n\}) p_t(m_t|\{\bar{x}_n\})] d\{\bar{x}_n\} \quad (62)$$

943 which suggests that the parameterized drift that minimizes the following minimization problem

$$a_t^{\theta^*}(m_t) = \arg \min_{\theta} \mathbb{E}_{q(\{\bar{x}_n\}) p_t(m_t|\{\bar{x}_n\})} [\|a_t^\theta(m_t) - a_t|\bar{x}_n(m_t|\{\bar{x}_n\})\|^2] \quad (63)$$

944 preserves the prescribed p_t . □

945 B.4 Proof of Theorem 3.5

946 **Definition B.5** (Markovian Projection of path measure). The Markovian Projection of \mathbb{P} is defined as
947 $\mathbb{P}^{\mathcal{M}} = \arg \min_{\mathbb{V} \in \mathcal{M}} KL(\mathbb{P}|\mathbb{V})$.

948 Intuitively, the Markovian Projection seeks the path measure that minimizes the variational distance
949 to \mathbb{P} . In other words, it seeks the closest Markovian path measure in the KL sense.

950 **Definition B.6** (Reciprocal Class and Projection). For multi-marginal path measures, we say that \mathbb{P}
951 is in the reciprocal class $\mathcal{R}(\mathbb{Q})$ of $\mathbb{Q} \in \mathcal{M}$ if

$$\mathbb{P} = \int \mathbb{Q}_{|\{x_n\}} dq(\{x_n\})$$

952 namely, it shares the same bridges with \mathbb{Q} . We define the *reciprocal projection* of \mathbb{P} as

$$\mathbb{P}^* = \text{proj}_{\mathcal{R}(\mathbb{Q})}(\mathbb{P}) := \arg \min_{\mathbb{T} \in \mathcal{R}(\mathbb{Q})} KL(\mathbb{P} \parallel \mathbb{T}).$$

953 Similarly, the Reciprocal Projection yields the closest reciprocal path measure in the KL sense.

954 **Lemma B.7.** Let \mathbb{P} be a Markov measure in the reciprocal class of $\mathbb{Q} \in \mathcal{M}$ such that $\int \mathbb{P}_n dv_n =$
 955 $q_n(x_n)$, for $n \in \{0, \dots, N\}$. Then, under some mild regularity assumptions on \mathbb{Q} , q_n , it is found that
 956 \mathbb{P} is equal to the unique multi-marginal the Schrödinger Bridge \mathbb{P}^{mmSB} .

957 *Proof.* First let us assume that $\text{KL}(\mathbb{P}|\mathbb{Q}) < \infty$, and that $\text{KL}(q_n|\int \mathbb{Q}_n dv_n) < \infty$ for $n \in \{0, \dots, N\}$.
 958 Assume $\mathbb{Q} \in \mathcal{M}$, then by (Theorem 2.10, Theorem 2.12 [Léonard \[2013\]](#)), it follows that the solution
 959 of the dynamic SB \mathbb{P} must also be a Markov measure. Finally, from the factorization of the KL, it
 960 holds that

$$\text{KL}(\mathbb{P}|\mathbb{Q}) = \text{KL}(\mathbb{P}_{\{x_n\}}|\mathbb{Q}_{\{x_n\}}) + \int \text{KL}(\mathbb{P}_{|\{x_n\}}|\mathbb{Q}_{|\{x_n\}}) d\mathbb{P}_{\{x_n\}} \quad (64)$$

961 which implies that $\text{KL}(\mathbb{P}_{\{x_n\}}|\mathbb{Q}_{\{x_n\}}) \leq \text{KL}(\mathbb{P}|\mathbb{Q})$ with equality (when $\text{KL}(\mathbb{P}|\mathbb{Q}) < \infty$) if and
 962 only if $\mathbb{P}_{|\{x_n\}} = \mathbb{Q}_{|\{x_n\}}$. Therefore, \mathbb{P}^* is the (unique) solution mmSB if and only if it disintegrates
 963 as above (Proposition 2.3 [Léonard \[2013\]](#)). \square

964 **Lemma B.8.** [Proposition 6 in [\[Shi et al., 2023\]](#)] Let $\mathbb{V} \in \mathcal{M}$ and $\mathbb{T} \in \mathcal{R}(\mathbb{Q})$ and . If $\text{KL}(\mathbb{P}|\mathbb{V}) < \infty$,
 965 and if $\text{KL}(\text{proj}_{\mathcal{M}}(\mathbb{P})|\mathbb{V}) < \infty$ we have

$$\text{KL}(\mathbb{P}|\mathbb{V}) = \text{KL}(\mathbb{P}|\text{proj}_{\mathcal{M}}(\mathbb{P})) + \text{KL}(\text{proj}_{\mathcal{M}}(\mathbb{P})|\mathbb{V}). \quad (65)$$

966 and if $\text{KL}(\mathbb{P}|\mathbb{T}) < \infty$, then

$$\text{KL}(\mathbb{P}|\mathbb{T}) = \text{KL}(\mathbb{P}|\text{proj}_{\mathcal{R}(\mathbb{Q})}(\mathbb{P})) + \text{KL}(\text{proj}_{\mathcal{R}(\mathbb{Q})}(\mathbb{P})|\mathbb{T}). \quad (66)$$

967 **Theorem B.9.** Assume that the conditions of Lemma B.7, and B.8 hold. Then, our iterative scheme
 968 admits a fixed point solution \mathbb{P}^* , i.e., $\text{KL}(\mathbb{P}^i|\mathbb{P}^*) \rightarrow 0$, and in particular, this fixed point coincides
 969 with the unique \mathbb{P}^{mmSB} .

970 *Proof.* We define the following path measures $\mathbb{V} = \text{proj}_{\mathcal{M}}(\mathbb{P})$, and $\mathbb{T} = \text{proj}_{\mathcal{R}(\mathbb{Q})}(\mathbb{V})$. Assume the
 971 conditions for Lemma B.8 hold for \mathbb{P} , \mathbb{V} and \mathbb{T} . Then for any arbitrary fixed point \mathbb{V}' , we can write

$$\text{KL}(\mathbb{V}^0|\mathbb{V}') = \text{KL}(\mathbb{V}^0|\mathbb{V}^1) + \text{KL}(\mathbb{V}^1|\mathbb{V}') = \sum_{i=0}^N \text{KL}(\mathbb{V}^i|\mathbb{V}') + \text{KL}(\mathbb{V}^i|\mathbb{V}') \quad (67)$$

972 Thus, it holds that $\text{KL}(\mathbb{V}^i|\mathbb{V}') < \infty$ for every iteration $i \in \mathbb{N}$. Similarly, for \mathbb{P} , and \mathbb{T} , we obtain
 973 $\text{KL}(\mathbb{P}^i|\mathbb{P}') < \infty$ and $\text{KL}(\mathbb{T}^i|\mathbb{T}') < \infty$ for each $i \in \mathbb{N}$ for any arbitrary fixed \mathbb{P}' and \mathbb{T}' .

974 Consequently, we define the following function

$$\Psi^i := \text{KL}(\mathbb{V}^i|\mathbb{V}') + \text{KL}(\mathbb{T}^i|\mathbb{T}') + \text{KL}(\mathbb{P}^i|\mathbb{P}'). \quad (68)$$

975 For two consecutive iterates i and $i+1$, we have

$$\begin{aligned} 976 & \bullet \Psi^i := \text{KL}(\mathbb{V}^i|\mathbb{V}') + \text{KL}(\mathbb{T}^i|\mathbb{T}') + \text{KL}(\mathbb{P}^i|\mathbb{P}') \\ 977 & \bullet \Psi^{i+1} := \text{KL}(\mathbb{V}^{i+1}|\mathbb{V}') + \text{KL}(\mathbb{T}^{i+1}|\mathbb{T}') + \text{KL}(\mathbb{P}^{i+1}|\mathbb{P}') \end{aligned}$$

978 Using Lemma B.8, we can rewrite Ψ^i as

$$\begin{aligned} \Psi^i &:= \text{KL}(\mathbb{V}^i|\mathbb{V}') + \text{KL}(\mathbb{T}^i|\mathbb{T}') + \text{KL}(\mathbb{P}^i|\mathbb{P}') \\ &= \text{KL}(\mathbb{V}^i|\mathbb{V}^{i+1}) + \text{KL}(\mathbb{T}^i|\mathbb{T}^{i+1}) + \text{KL}(\mathbb{P}^i|\mathbb{P}^{i+1}) + \text{KL}(\mathbb{V}^{i+1}|\mathbb{V}') + \text{KL}(\mathbb{T}^{i+1}|\mathbb{T}') + \text{KL}(\mathbb{P}^{i+1}|\mathbb{P}') \\ &= \text{KL}(\mathbb{V}^i|\mathbb{V}^{i+1}) + \text{KL}(\mathbb{T}^i|\mathbb{T}^{i+1}) + \text{KL}(\mathbb{P}^i|\mathbb{P}^{i+1}) + \Psi^{i+1} \end{aligned}$$

979 Now, we take the sum of this telescoping series and obtain

$$\Psi^0 - \Psi^\infty \geq \sum_{i=0}^{\infty} \text{KL}(\mathbb{V}^i|\mathbb{V}^{i+1}) + \text{KL}(\mathbb{T}^i|\mathbb{T}^{i+1}) + \text{KL}(\mathbb{P}^i|\mathbb{P}^{i+1}) \quad (69)$$

980 Note that Ψ^0 and Ψ^∞ are finite (with $\Psi^0 \geq \Psi^\infty$), since $\text{KL}(\mathbb{P}^i|\mathbb{P}') < \infty$, $\text{KL}(\mathbb{V}^i|\mathbb{V}') < \infty$ and
 981 $\text{KL}(\mathbb{T}^i|\mathbb{T}') < \infty$ for every iteration $i \in \mathbb{N}$. Therefore, since we also have $\text{KL}(\mathbb{P}^i|\mathbb{P}^{i+1}) \geq 0$,
 982 $\text{KL}(\mathbb{V}^i|\mathbb{V}^{i+1}) \geq 0$ and $\text{KL}(\mathbb{T}^i|\mathbb{T}^{i+1}) \geq 0$, we get that

- 983 • $\lim_{i \rightarrow \infty} \text{KL}(\mathbb{P}^i | \mathbb{P}^{i+1}) \rightarrow 0$,
- 984 • $\lim_{i \rightarrow \infty} \text{KL}(\mathbb{V}^i | \mathbb{V}^{i+1}) \rightarrow 0$,
- 985 • $\lim_{i \rightarrow \infty} \text{KL}(\mathbb{T}^i | \mathbb{T}^{i+1}) \rightarrow 0$.

986 Hence the iterates \mathbb{P}^i , \mathbb{V}^i , and \mathbb{T}^i converge to some fixed points $\mathbb{P}^i \rightarrow \mathbb{P}^*$, $\mathbb{V}^i \rightarrow \mathbb{V}^*$, and $\mathbb{T}^i \rightarrow \mathbb{T}^*$,
 987 as $i \rightarrow \infty$.

988 From the factorization of the KL divergence, we have for consecutive projections of our algorithm:

$$\begin{aligned} \text{KL}(\mathbb{T}^{(i)} | \mathbb{P}^*) &= \text{KL}(\mathbb{T}_{\{x_n\}}^{(i)} | \mathbb{P}_{\{x_n\}}^*) + \mathbb{E}_{\mathbb{T}_{\{x_n\}}^{(i)}} \left[\text{KL}(\mathbb{T}_{\{x_n\}}^{(i)} | \mathbb{P}_{\{x_n\}}^*) \right] \\ &= \text{KL}(\mathbb{T}_{\{x_n\}}^{(i)} | \mathbb{P}_{\{x_n\}}^*) \end{aligned} \quad (70)$$

989 since the bridges of \mathbb{T} are the same with \mathbb{P} , and \mathbb{Q} , i.e., $\mathbb{T}_{\{x_n\}}^{(i)} = \mathbb{P}_{\{x_n\}}^* = \mathbb{Q}_{\{x_n\}}$. Then,

$$\begin{aligned} \text{KL}(\mathbb{V}^i | \mathbb{P}^*) &= \text{KL}(\mathbb{V}_{\{x_n\}}^i | \mathbb{P}_{\{x_n\}}^*) + \mathbb{E}_{\mathbb{V}_{\{x_n\}}^i} \left[\text{KL}(\mathbb{V}_{\{x_n\}}^i | \mathbb{P}_{\{x_n\}}^*) \right] \\ &\geq \text{KL}(\mathbb{T}_{\{x_n\}}^{(i+1)} | \mathbb{P}_{\{x_n\}}^*) \end{aligned} \quad (71)$$

990 since the coupling after the Markovian projection at iteration i remains the same for the reciprocal
 991 path measure at iteration $i + 1$, namely $\mathbb{V}_{\{x_n\}}^{(i)} = \mathbb{T}_{\{x_n\}}^{(i+1)}$. Therefore, we can deduce

$$\text{KL}(\mathbb{V}^i | \mathbb{P}^*) \geq \text{KL}(\mathbb{T}^{i+1} | \mathbb{P}^*). \quad (72)$$

992 We further assume that $\text{KL}(\mathbb{T}^0 | \mathbb{P}^*) < \infty$, $\text{KL}(\mathbb{V}^0 | \mathbb{P}^*) < \infty$. Therefore, the iterations of
 993 Eq. 70 and 71, yield $\text{KL}(\mathbb{T}^{(i)} | \mathbb{P}^*) \geq \text{KL}(\mathbb{V}^i | \mathbb{P}^*) \geq \text{KL}(\mathbb{T}^{i+1} | \mathbb{P}^*)$ for $i \geq 0$, implying con-
 994 vergence, since it is non-increasing and bounded below. Applying Lemma B.8, we obtain
 995 $\lim_{i \rightarrow \infty} (\text{KL}(\mathbb{T}^i | \mathbb{P}^*) - \text{KL}(\mathbb{V}^i | \mathbb{P}^*)) = \lim_{i \rightarrow \infty} \text{KL}(\mathbb{T}^i | \mathbb{V}^i) = 0$. By definition of the lower semi-
 996 continuity of the KL divergence, we have $\text{KL}(\mathbb{V}^* | \mathbb{T}^*) \leq \liminf_{k \rightarrow \infty} \text{KL}(\mathbb{V}^{i_{j^k}} | \mathbb{T}^{i_{j^k}})$. Additionally,
 997 by the definition of the KL divergence, we have $0 \leq \text{KL}(\mathbb{V}^* | \mathbb{T}^*)$. Finally, it also holds that
 998 $\liminf_{k \rightarrow \infty} \text{KL}(\mathbb{V}^{i_{j^k}} | \mathbb{T}^{i_{j^k}}) = 0$. Combining all three claims we have

$$0 \leq \text{KL}(\mathbb{V}^* | \mathbb{T}^*) \leq \liminf_{k \rightarrow \infty} \text{KL}(\mathbb{V}^{i_{j^k}} | \mathbb{T}^{i_{j^k}}) = 0.$$

999 Therefore, $\mathbb{V}^* = \mathbb{T}^*$, which also means that $\mathbb{P}^* \in \mathcal{M} \cap \mathcal{R}(\mathbb{Q})$ [Shi et al., 2023]. Also, by construction,
 1000 all the iterates of \mathbb{P}^i satisfy the positional marginal constraints $\int \mathbb{P}_n^i dv_n = q_n$, hence also $\int \mathbb{P}_n^* dv_n =$
 1001 q_n . Therefore, by Lemma B.7, \mathbb{P}^* is the unique multi-marginal Schrödinger bridge \mathbb{P}^{mmSB} .

1002 □

1003 C Gaussian Path

1004 The scalability of our framework is based on the capacity to efficiently sample from the conditional
 1005 gaussian path induced by the solution of the 3MBB path in Eq 6. Let us define the marginal path
 1006 p_t as a mixture of bridges $p_t(m_t) = \int p_{t|\{\bar{x}_n\}}(m_t|\{\bar{x}_n : t_n \geq t\}) dq(\{x_n\})$, where $p_{t|\{\bar{x}_n\}}(m_t|\{\bar{x}_n :$
 1007 $t_n \geq t\})$ is the conditional probability path associated with the solution of the 3MBB path in Eq 6.
 1008 The linearity of the system implies that we can efficiently sample $m_t = [x_t, v_t]$, $\forall t \in [0, T]$, from the
 1009 conditional probability path $p_{t|\{\bar{x}_n\}} = \mathcal{N}(\mu_t, \Sigma_t)$, as the mean vector μ_t and the covariance matrix
 1010 Σ_t have analytic solutions [Särkkä and Solin, 2019].

1011 Let us recall the optimal control formulation for the n^{th} segment.

$$u_t^* = \begin{bmatrix} 0 & 0 \\ C_1^{(n)}(t) & C_2^{(n)}(t) \end{bmatrix} \begin{bmatrix} \mathbf{x}_t \\ \mathbf{v}_t \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ C_1^{(n)}(t) & C_3^{(n)}(t) \end{bmatrix} \begin{bmatrix} -\bar{x}_{n+1} \\ \sum_{j=n+1}^N \lambda_j \bar{x}_j \end{bmatrix}, \forall t \in [t_n, t_{n+1}] \quad (73)$$

1012 Thus, the augmented SDE for the corresponding is written as

$$dm_t = \left(\begin{bmatrix} 0 & 1 \\ C_1^{(n)}(t) & C_2^{(n)}(t) \end{bmatrix} \begin{bmatrix} \mathbf{x}_t \\ \mathbf{v}_t \end{bmatrix} + \begin{bmatrix} 0 \\ -C_1^{(n)}(t)\bar{x}_{n+1} + (C_3^{(n)}(t)) \sum_{j=n+1}^N \lambda_j^{(n)} \bar{x}_j \end{bmatrix} \right) dt + \mathbf{g} dW_t \quad (74)$$

1013 where $C_1^{(n)}(t), C_2^{(n)}(t), C_3^{(n)}(t)$ are the segment specific functions, and $\lambda_j^{(n)}$ the coefficients for the
 1014 future pinned points. To find the expressions for the mean and the covariance, we follow [Särkkä and
 1015 Solin, 2019], and consider the following ODEs for μ_t , and Σ_t respectively:

$$\begin{aligned} \dot{\mu}_t &= \begin{bmatrix} 0 & 1 \\ C_1^{(n)}(t) & C_2^{(n)}(t) \end{bmatrix} \mu_t + \begin{bmatrix} 0 \\ -C_1^{(n)}(t)\bar{x}_{n+1} + C_3^{(n)}(t) \sum_{j=n+1}^N \lambda_j^{(n)} \bar{x}_j \end{bmatrix} \\ \dot{\Sigma}_t &= \begin{bmatrix} 0 & 1 \\ C_1^{(n)}(t) & C_2^{(n)}(t) \end{bmatrix} \Sigma_t + \begin{bmatrix} 0 & 1 \\ C_1^{(n)}(t) & C_2^{(n)}(t) \end{bmatrix} \Sigma_t^\top + \mathbf{g} \mathbf{g}^\top \end{aligned} \quad (75)$$

1016 **Mean ODEs** If we explicitly write the mean ODE system, we obtain the following two ODEs

$$\begin{aligned} \dot{\mu}_x &= \mu_v \\ \dot{\mu}_v &= C_1^{(n)}(t)\mu_x + C_2^{(n)}(t)\mu_v - C_1^{(n)}(t)\bar{x}_{n+1} + C_3^{(n)}(t) \sum_{j=n+1}^N \lambda_j^{(n)} \bar{x}_j \end{aligned} \quad (76)$$

1017 which corresponds to the following second-order ODE

$$\ddot{\mu}_x - C_2^{(n)}(t)\dot{\mu}_x + C_1^{(n)}(t)\mu_x = C_1^{(n)}(t)\bar{x}_{n+1} + C_3^{(n)}(t) \sum_{j=n+1}^N \lambda_j^{(n)} \bar{x}_j \quad (77)$$

1018 This ODE is then solved using an ODE solver software for the corresponding functions $C^{(n)}$ of the
 1019 respective segment n : $t \in [t_n, t_{n+1})$.

1020 **Covariance ODEs** If we explicitly write the mean ODE system, we obtain the following two ODEs

$$\begin{aligned} \dot{\Sigma}_{xx} &= 2\Sigma_{xv} \\ \dot{\Sigma}_{xv} &= C_1^{(n)}(t)\Sigma_{xx} + C_2^{(n)}(t)\Sigma_{xv} + \Sigma_{vv} \\ \dot{\Sigma}_{vv} &= 2C_1^{(n)}(t)\Sigma_{xv} + 2C_2^{(n)}(t)\Sigma_{vv} + \sigma^2 \end{aligned} \quad (78)$$

1021 which corresponds to the following third-order ODE

$$\frac{1}{2} \ddot{\Sigma}_{xx} - \frac{3}{2} C_2^{(n)}(t) \ddot{\Sigma}_{xx} + (C_2^{(n)}(t)^2 - 2C_1^{(n)}(t) - \frac{1}{2} \dot{C}_2^{(n)}(t)) \dot{\Sigma}_{xx} + (2C_2^{(n)}(t)C_1^{(n)}(t) - \dot{C}_1^{(n)}(t)) \Sigma_{xx} = 0 \quad (79)$$

1022 This equation, however, is hard to solve even using software packages. For this reason, we integrate
 1023 the covariance ODEs using Euler integration, once at the beginning of our training. This procedure
 1024 can be solved once and can be applied for any fixed coupling, during the matching, since the system
 1025 of ODEs in Eq. (78) does not depend on any points \bar{m}_n , but its sole dependence is on time. This
 1026 suggests that the computational overhead from simulating the covariance ODEs is negligible.

1027 Given the expressions of $\mu_t = \begin{bmatrix} \mu_t^x \\ \mu_t^v \end{bmatrix}$, and $\Sigma_t = \begin{bmatrix} \Sigma_t^{xx} & \Sigma_t^{xv} \\ \Sigma_t^{xv} & \Sigma_t^{vv} \end{bmatrix}$, one can obtain m_t through

$$m_t = \begin{bmatrix} X_t \\ V_t \end{bmatrix} = \begin{bmatrix} \mu_t^x \\ \mu_t^v \end{bmatrix} + \begin{bmatrix} L_t^{xx} \epsilon_0 \\ L_t^{xv} \epsilon_0 + L_t^{vv} \epsilon_1 \end{bmatrix} \quad (80)$$

1028 where the matrix $L_t = \begin{bmatrix} L_t^{xx} & L_t^{xv} \\ L_t^{xv} & L_t^{vv} \end{bmatrix}$ is computed following the Cholesky decomposition to the
 1029 covariance matrix, and $\epsilon = \begin{bmatrix} \epsilon_0 \\ \epsilon_1 \end{bmatrix} \sim \mathcal{N}(0, \mathbf{I}_{2d})$.

D Extended Related Works

Schrödinger Bridge Recently, in generative modeling there has been a surge of principled approaches that stem from Optimal Transport [Villani et al., 2009]. The most prominent problem formulation has been the Schrödinger Bridge (SB; Schrödinger [1931]). In particular, SB gained significant popularity in the realm of generative modeling following advancements proposing a training scheme based on the Iterative Proportional Fitting (IPF), a continuous state space extension of the Sinkhorn algorithm to solve the dynamic SB problem [De Bortoli et al., 2021, Vargas et al., 2021]. Notably, SB generalizes standard diffusion models transporting data between arbitrary distributions π_0, π_1 with fully nonlinear stochastic processes, seeking the unique path measure that minimizes the kinetic energy. The Schrödinger Bridge [Schrödinger, 1931] in the path measure sense is concerned with finding the optimal measure \mathbb{P}^* that minimizes the following optimization problem

$$\min_{\mathbb{P}} KL(\mathbb{P}|\mathbb{Q}), \quad \mathbb{P}_0 = \pi_0, \mathbb{P}_1 = \pi_1 \quad (81)$$

where \mathbb{Q} is a Markovian reference measure. Hence the solution of the dynamic SB \mathbb{P}^* is considered to be the closest path measure to \mathbb{Q} . Another formulation of the dynamic SB crucially emerges by applying the Girsanov theorem framing the problem as a Stochastic Optimal Control (SOC) Problem [Chen et al., 2016, 2021].

$$\min_{u_t, p_t} \int_0^1 \mathbb{E}_{p_t} [\|u_t\|^2] dt \quad \text{s.t.} \quad \frac{\partial p_t}{\partial t} = -\nabla \cdot (u_t p_t) + \frac{\sigma^2}{2} \Delta p_t, \quad \text{and} \quad p_0 = \pi_0, \quad p_1 = \pi_1 \quad (82)$$

Finally, note that the static SB is equivalent to the entropy regularized OT formulation [Pavon et al., 2021, Nutz, 2021, Cuturi, 2013].

$$\min_{\pi \in \Pi(\pi_0, \pi_1)} \int_{\mathbb{R}^d \times \mathbb{R}^d} \|x_0 - x_1\|^2 d\pi(x_0, x_1) + \epsilon KL(\pi|\pi_0 \otimes \pi_1) \quad (83)$$

This regularization term enabled efficient solution through the Sinkhorn algorithm and has presented numerous benefits, such as smoothness, and other statistical properties [Ghosal et al., 2022, Léger, 2021, Peyré et al., 2019].

Bridge Matching Peluchetti [2023] first proposed the Markovian projection to propose Bridge matching, while Liu et al. [2022a] employed it to learn representations in constrained domains. The Bridge matching objective offers a computationally efficient alternative but requires additional assumptions. To this front, Action Matching Neklyudov et al. [2023] presents a general matching method with the least assumptions, at the expense of being unfavorable to scalability. Additionally, recent advances have introduced more general frameworks for conditional generative modeling. Denoising Diffusion Bridge Models (DDBMs) extend traditional diffusion models to handle arbitrary source and target distributions by learning the score of a diffusion bridge, thereby unifying and generalizing methods such as score-based diffusion and flow matching [Zhou et al., 2023]. Similarly, the stochastic interpolant framework [Albergo et al., 2023] integrates flow- and diffusion-based approaches by defining continuous-time stochastic processes that interpolate between distributions. These interpolants achieve exact bridging in finite time by introducing an auxiliary latent variable, offering flexible control over the interpolation path.

Recently, these matching frameworks have been employed to solve the SB problem. DSBM ([Shi et al., 2023]) employs Iterative Markovian Fitting (IMF) to obtain the Schrodinger Bridge solution, while De Bortoli et al. [2023] explores flow and bridge matching processes, proposing a modification to preserve coupling information, demonstrating efficiency in learning mixtures of image translation tasks. $SF^2 - M$ [Tong et al., 2023a] provides a simulation-free objective for inferring stochastic dynamics, demonstrating efficiency in solving Schrödinger bridge problems. GSBM Liu et al. [2024] presents a framework for solving distribution matching to account for task-specific state costs. While these methods aim to identify the optimal coupling, [Somnath et al., 2023] and [Liu et al., 2023] propose Bridge Matching algorithms between a priori coupled data, namely the pairing between clean and corrupted images or pairs of biological data from the static Schrödinger Bridge. Lastly, works that aim to improve the efficiency of these matching frameworks by introducing a light solver for implementing optimal matching using Gaussian mixture parameterization [Gushchin et al., 2024a].

Flow Matching In parallel, there have been methodologies that employ deterministic dynamics. Lipman et al. [2022] introduces the deterministic counterpart of Bridge Matching; Flow Matching (FM) for training Continuous Normalizing Flows (CNFs; Chen et al. [2018]) using fixed conditional probability paths. Further developments include Conditional Flow Matching (CFM) offers a stable regression objective for training CNFs without requiring Gaussian source distributions or density evaluations [Tong et al., 2023b], Metric Flow Matching (MFM), which learns approximate geodesics on data manifolds [Kapusniak et al., 2024], and Flow Matching in Latent Space, which improves computational efficiency for high-resolution image synthesis [Dao et al., 2023]. Finally, CFM retrieves exactly the first iteration of the Rectified Flow Liu et al. [2022b], which is an iterative approach for learning ODE models to transport between distributions.

Multi-Marginal Among the advancements of Flow matching models was the introduction of the Multi-Marginal Flow matching framework [Rohbeck et al., 2024]. Similarly to our approach, they proposed a simulation-free training approach, leverages cubic spline-based flow interpolation and classifier-free guidance across time and conditions. TrajectoryNet [Tong et al., 2020] and MIOFlow [Huguet et al., 2022] combine Optimal Transport with Continuous Normalizing Flows [Chen et al., 2018] and manifold embeddings, respectively, to model non-linear continuous trajectories through multiple points. In the stochastic realm, recent works have proposed the mmSB training through extending Iterative Proportional Fitting - a continuous extension of the Sinkhorn algorithm to solve the dynamic SB problem [De Bortoli et al., 2021]- to phase space and adapting the Bregman iterations [Chen et al., 2023a]. Another approach alternates between learning piecewise SBs on the unobserved trajectories and refining the best guess for the dynamics within the specified reference class [Shen et al., 2024]. More recently, modeling the reference dynamics as a special class of smooth Gaussian paths was shown to achieve more regular and interpretable trajectories [Hong et al., 2025]. Furthermore, the multi-marginal problem has been recently addressed by Deep Momentum Multi-Marginal Schrödinger Bridge [DMSB; Chen et al. [2023a]] proposed to solve the mmSB in phase space via adapting the Bregman iterations. More recently, an iterative method for solving the mmSB proposed learning piecewise SB dynamics within a preselected reference class [Shen et al., 2024]. In contrast, modeling the reference dynamics as smooth Gaussian paths was shown to achieve more temporally coherent and smooth trajectories [Hong et al., 2025], though the belief propagation prohibits scaling in high dimensions.

E Additional Details on Experiments

E.1 General Information

In this section, we revisit our experimental results to evaluate the performance of 3MSBM on a variety of trajectory inference tasks, such as Lotka-Volterra, ocean current in the Gulf of Mexico, single-cell sequencing, and the Beijing air quality data. We compared against state-of-the-art methods explicitly designed to incorporate multi-marginal settings, such Deep Momentum Multi-Marginal Schrödinger Bridge (DMSB; Chen et al. [2023a]), Schrodinger Bridge with Iterative Reference Refinement (SBIRR; Shen et al. [2024]), smooth Schrodinger Bridges (smoothSB; Hong et al. [2025]), and Multi-Marginal Flow Matching (MMFM; Rohbeck et al. [2024]). We used the official implementations of all compared methods, with default hyperparameters. For all experiments with our 3MSBM, we employed the ResNet architectures from Chen et al. [2023b], Dockhorn et al. [2021]. We used the AdamW optimizer and applied Exponential Moving Averaging with a decay rate of 0.999. All results are averaged over 5 random seeds, with means and standard deviations reported in Section 4 and the tables below. Experiments were run on an RTX 4090 GPU with 24 GB of VRAM.

E.2 Lotka-Volterra

We first consider a synthetic dataset generated by the Lotka–Volterra (LV) equations [Goel et al., 1971], which model predator-prey interactions through coupled nonlinear dynamics. We used the dataset from [Shen et al., 2024] with the 5 training and 4 validation time points, with 50 observations per time point. In particular, the generated dataset consists of 9 marginals in total; the even-numbered indices are used to train the model (i.e., t_0, t_2, t_4, t_6, t_8), and the remainder of the time points are used to assess the efficacy of our model to impute and infer the missing time points. In this experiment, we benchmarked 3MSBM against SBIRR, MMFM, and Smooth SB. Table 7 reports the performance

of each method in \mathcal{W}_1 distance from the validation points. The hyperparameter selection for the LV with our method were: the diffusion coefficient was set to $\sigma = 0.3$, and the learning rate was 10^{-4} .

Table 7: Mean and SD over 5 seeds for \mathcal{W}_1 distances at left-out marginals on the LV data for SBIRR, MMFM, Smooth SB, and 3MSBM (Lower is better).

Method	$\mathcal{W}_1 t_1$	$\mathcal{W}_1 t_3$	$\mathcal{W}_1 t_5$	$\mathcal{W}_1 t_7$
SBIRR	0.19 ± 0.02	0.20 ± 0.02	0.30 ± 0.03	0.48 ± 0.03
MMFM	0.24 ± 0.02	0.43 ± 0.04	0.57 ± 0.05	0.75 ± 0.04
Smooth SB	0.29 ± 0.01	0.27 ± 0.03	0.22 ± 0.04	0.68 ± 0.03
3MSBM (ours)	0.23 ± 0.03	0.18 ± 0.02	0.12 ± 0.03	0.35 ± 0.03

E.3 Gulf of Mexico

Subsequently, we evaluate the efficacy of our model to infer the missing time points in a real-world multi-marginal dataset. The dataset contains ocean-current snapshots of the velocity field around a vortex in the Gulf of Mexico (GoM). Similarly to the LV dataset, we used the big vortex dataset provided in [Shen et al., 2024], consisting of 300 samples across 5 training and 4 validation times. More explicitly, out of the total 9 marginals, the even-indexed time points (i.e., t_0, t_2, t_4, t_6, t_8) are used for training, and the remaining are left out to evaluate the model’s ability to impute and infer missing temporal states. We compared our 3MSBM against SBIRR, MMFM, and Smooth SB for this experiment. Table 8 reports the performance from the validation points using the \mathcal{W}_1 distance from the left-out marginals. The hyperparameters used for the GoM experiment with our method were: a batch size of 32 for the matching, the diffusion coefficient was set to $\sigma = 0.3$, and the learning rate was set equal to $2 \cdot 10^{-4}$.

Table 8: Mean and SD over 5 seeds for \mathcal{W}_1 distances at left-out marginals on the GoM data for SBIRR, MMFM, Smooth SB, and 3MSBM (Lower is better).

Method	$\mathcal{W}_1 t_1$	$\mathcal{W}_1 t_3$	$\mathcal{W}_1 t_5$	$\mathcal{W}_1 t_7$
SBIRR	0.28 ± 0.05	0.15 ± 0.03	0.11 ± 0.02	0.15 ± 0.02
MMFM	0.33 ± 0.03	0.38 ± 0.04	0.22 ± 0.03	0.29 ± 0.04
Smooth SB	0.16 ± 0.02	0.27 ± 0.03	0.21 ± 0.03	0.56 ± 0.03
3MSBM (ours)	0.17 ± 0.03	0.21 ± 0.01	0.09 ± 0.02	0.12 ± 0.03

E.4 Beijing air quality

We revisit our experiments using the Beijing multi-site air quality data set [Chen, 2017]. This dataset consists of hourly air pollutant data from 12 air-quality monitoring sites across Beijing. We focus on PM2.5 data, an indicator monitoring the density of particles smaller than 2.5 micrometers, from January 2013 to January 2015. We focused on a single monitoring site and aggregated the measurements collected within the same month. To introduce temporal separation between observations, we selected measurements from every other month, resulting in 13 temporal snapshots. For the imputation task, we omitted the data at t_2, t_5, t_8 , and t_{11} , while the remaining snapshots formed the training set. Table 10 shows the performance of each method in the \mathcal{W}_2 distance from the validation time points, benchmarking our 3MSBM method against MMFM with cubic splines. The hyperparameters used for the Beijing air quality experiment with our method were: a total number of samples of 1000 were used, with a batch size of 64 for the matching, the diffusion coefficient was set to $\sigma = 0.2$, and the learning rate was set to $5 \cdot 10^{-5}$.

Table 9: \mathcal{W}_2 distances at left-out marginals, on the Beijing air quality data for MMFM, DMSB, and 3MSBM (Lower is better).

Method	$\mathcal{W}_2 t_2$	$\mathcal{W}_2 t_5$	$\mathcal{W}_2 t_8$	$\mathcal{W}_2 t_{11}$
MMFM	26.42 ± 4.01	29.95 ± 2.15	43.56 ± 3.12	49.98 ± 4.93
3MSBM(ours)	12.87 ± 1.87	79.36 ± 12.65	27.89 ± 5.88	32.26 ± 4.15

Table 11: Mean and SD over 5 seeds for the Maximum Mean Discrepancy (MMD) and SWD at the left-out marginals t_1, t_3 on Embryoid Body (EB) dataset for the 5-marginal case for SBIRR, MMFM, DMSB, and 3MSBM (Lower is better).

Method	MMD t_1	SWD t_1	MMD t_3	SWD t_3
SBIRR	0.29 \pm 0.03	0.37 \pm 0.03	0.34 \pm 0.04	0.45 \pm 0.04
MMFM	0.33 \pm 0.05	0.39 \pm 0.04	0.48 \pm 0.04	0.45 \pm 0.04
DMSB	0.28 \pm 0.04	0.32 \pm 0.04	0.24 \pm 0.02	0.34 \pm 0.04
3MSBM (ours)	0.17 \pm 0.03	0.28 \pm 0.03	0.24 \pm 0.03	0.23 \pm 0.03

Table 12: Mean and SD over 5 seeds for the Maximum Mean Discrepancy (MMD) and SWD at the left-out marginals t_1, t_2 on Embryoid Body (EB) dataset for the 4-marginal case for SBIRR, MMFM, DMSB, and 3MSBM (Lower is better).

Method	MMD t_1	SWD t_1	MMD t_2	SWD t_2
SBIRR	0.37 \pm 0.05	0.37 \pm 0.04	0.45 \pm 0.04	0.61 \pm 0.05
MMFM	0.39 \pm 0.04	0.39 \pm 0.04	0.45 \pm 0.03	0.63 \pm 0.04
DMSB	0.60 \pm 0.04	0.35 \pm 0.03	0.50 \pm 0.02	0.57 \pm 0.04
3MSBM (ours)	0.28 \pm 0.04	0.37 \pm 0.04	0.30 \pm 0.03	0.45 \pm 0.03

Table 10: MMD distance at left-out marginals, on the Beijing air quality data for MMFM, DMSB, and 3MSBM (Lower is better).

Method	MMD t_2	MMD t_5	MMD t_8	MMD t_{11}
MMFM	0.44 \pm 0.06	0.56 \pm 0.13	0.59 \pm 0.04	0.55 \pm 0.04
3MSBM(ours)	0.35 \pm 0.09	0.85 \pm 0.07	0.28 \pm 0.04	0.32 \pm 0.05

E.5 Single sequencing

Lastly, we revisit our experiments on the Embryoid Body (EB) stem cell differentiation dataset, which captures cell progression across 5 stages over a 27-day period. Following the setup in Section 4.4, we used the preprocessed data from [Tong et al., 2020, Moon et al., 2019], embedded in a 100-dimensional PCA feature space. Cell snapshots were collected at five discrete intervals: $t_0 \in [0, 3]$, $t_1 \in [6, 9]$, $t_2 \in [12, 15]$, $t_3 \in [18, 21]$, $t_4 \in [24, 27]$. Below, we first perform an evaluation on the full 5 marginals, and use a lower-dimensional embedding, specifically 5-feature space PCA. Subsequently, we reduce the number of marginals from 5 to 4 by using the first four time points and experimented with a 50-dimensional PCA spaces. We compared 3MSBM against SBIRR, MMFM, and DMSB, using Sliced Wasserstein Distance (SWD) and Maximum Mean Discrepancy (MMD) as evaluation metrics. Consistent with earlier experiments, we evaluated both the quality of imputed marginals and the fidelity in preserving training marginals. The hyperparameters used for every EB experiment with our method were: a total number of samples of 2000 were used, with a batch size of 64 for the matching, the diffusion coefficient was set to $\sigma = 0.1$, and the learning rate was set to 10^{-4} . This underlines our model’s capacity to outperform state-of-the-art algorithms in a variety of settings and dimensionalities with minimal hyperparameter tuning.

E.6 Ablation study on σ

In stochastic optimal control [Theodorou et al., 2010], the value of σ plays a crucial role in representing the uncertainty from the environment or the error in applying the control. As a result, the optimal control policy can vary significantly with different degrees of noise. Figure 12 demonstrates the performance of our 3MSBM with respect to varying noise in the EB and GoM datasets. We observe consistent performance in the training marginals across all tested values of σ , whereas for the points in the validation set, increasing σ up to a point is deemed beneficial as it improves performance. Sample trajectories in GoM in Figure 11 further verify this trend. Low noise values (e.g. $\sigma = 0.05$) cause the trajectories to be overly tight, whereas at high noise (e.g. $\sigma = 1.0$), the trajectories become

1179 overly diffuse. On the other hand, moderate noise values (e.g. $\sigma = 0.4$) achieve a good balance,
 1180 enabling well-spread trajectories matching the validation marginals.

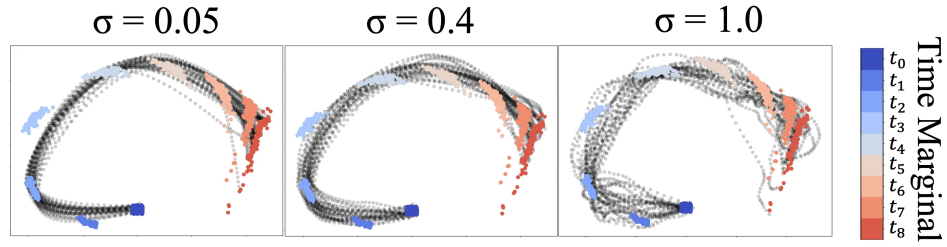


Figure 11: Comparison of the trajectories inferred on the Gulf Mexico current dataset for different values of σ

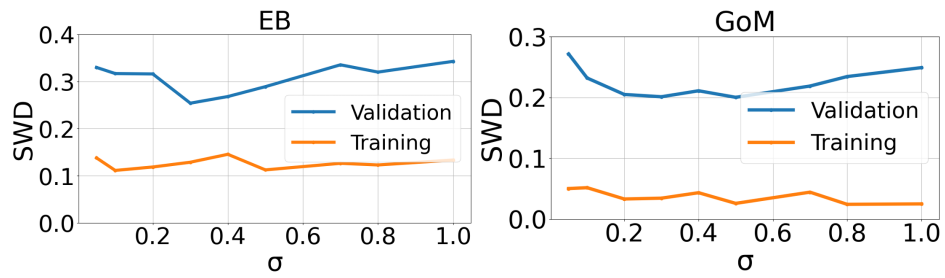


Figure 12: SWD from the marginals in the Validation and Training set for varying values of sigma on EB and GoM