Chapter 7: Flow Matching

This chapter introduces *flow matching*, a closely related but conceptually distinct framework compared to diffusion models. Instead of starting from coupled forward and backward stochastic processes in diffusion models, flow matching begins by specifying a *probability path* that interpolates between an easy-to-sample distribution and the data distribution, and then learns a time-dependent vector field that transports samples along that path.

The goal of this chapter is to present the minimal mathematical backbone of flow matching. We first review continuous-time flows and the continuity equation. We then introduce probability paths and explain how to define target velocity fields through conditional constructions. This leads to the flow matching objective and its conditional version. Finally, we show how diffusion paths arise as special Gaussian paths, clarifying the relation between flow matching and the diffusion models developed earlier in the course. A more comprehensive introduction to flow matching and more advanced contents such as discrete flow matching and generator matching can be found in [2].

1 Continuous-Time Flows

In diffusion models, sampling starts from an easy-to-sample distribution, typically Gaussian noise, and then applies a learned backward dynamics that gradually transforms noise into a data sample. Although this dynamics is derived from reversing a forward noising process, the generative objective is more basic: one seeks a continuous transformation that carries a simple reference distribution to the data distribution.

This viewpoint motivates the central object in flow matching. Rather than emphasizing the particular stochastic forward process used in diffusion models, flow matching focuses directly on constructing a family of intermediate distributions that connects the base distribution to the target data distribution. Such a family is called a probability path.

However, a probability path is not a generative model yet. To turn such a probability path into a generative model, we need a dynamical process whose distribution at each time matches the prescribed intermediate law. Flow matching adopts a deterministic continuous-time dynamics for this purpose, specified by a time-dependent velocity field. This dynamics can be described from two complementary viewpoints: at the sample level, it defines trajectories through an ordinary differential equation; at the distribution level, it determines the evolution of densities through the continuity equation. We next introduce these two descriptions.

1.1 Velocity Fields and Flows

Throughout this chapter, we consider a unit time interval $[0, 1]$ for convenience. A vector field is denoted as $u : \mathbb{R}^d \times [0, 1] \to \mathbb{R}^d$, which dictates the evolution of a dynamical system described by an Ordinary Differential Equation (ODE):

$$\frac{\mathrm{d}}{\mathrm{d}t}\psi(x, t) = u(\psi(x, t), t), \qquad \psi(x, 0) = x. \tag{1.1}$$

For each initial point $x \in \mathbb{R}^d$, the solution $t \mapsto \psi(x, t)$ traces a trajectory driven by the vector field $u(\cdot, t)$. The map $\psi : \mathbb{R}^d \times [0, 1] \to \mathbb{R}^d$ is called the *flow*.

Let the initial distribution be $X_0 \sim P_0$ (P_0 is simple), then the random variable

$$X_t = \psi(X_0, t)$$

has a marginal distribution

$$P_t = (\psi(\cdot, t))_\sharp P_0,$$

the pushforward of P_0 under $\psi(\cdot, t)$. A flow matching model based on (1.1) aims to choose u so that the terminal distribution $P_1 = (\psi(\cdot, 1))_\sharp P_0$ matches the data distribution P_{data}. From a distribution viewpoint, flow matching model constructs a *probability path* $\{P_t = (\psi(\cdot, t))_\sharp P_0\}_{t=0}^1$ using the velocity field u.

1.2 Continuity Equation

Under regularity conditions (e.g., Lipschitz continuity), the density p_t of marginal distribution P_t exists and evolves according to the *continuity equation*:

$$\frac{\partial}{\partial t} p_t(x) + \nabla \cdot \big(p_t(x) u_t(x)\big) = 0, \tag{1.2}$$

where $\nabla \cdot \big(p_t(x) u_t(x)\big)$ denotes the divergence with respect to x. Equation (1.2) expresses conservation of probability mass: the vector field u_t moves particles, and the density changes exactly according to this motion. Thus, if we can learn a vector field whose induced marginal densities $\{p_t\}_{t \in [0,1]}$ connect a simple base distribution P_0 to P_{data}, then solving the ODE from $t = 0$ to $t = 1$ yields a sampler for new data generation.

1.3 Sampling by ODE Integration

Under the assumption that u is perfectly chosen so that the terminal distribution is $P_1 = P_{\text{data}}$. Then we can generate samples from P_{data} by simulating the ODE (1.1). Conceptually, we

1. sample initial value x_0 for $X_0 \sim P_0$, which is typically a standard Gaussian;
2. numerically solve the ODE
$$\frac{\mathrm{d}}{\mathrm{d}t} X_t = u(X_t, t)$$
up to time $t = 1$ with initial condition $X_0 = x_0$;
3. output X_1.

In practice, one uses a numerical ODE solver such as Euler, midpoint, or higher-order adaptive solvers. Compared with diffusion samplers based on many stochastic denoising steps, flow matching models emphasize deterministic transport under a velocity field.

2 Probability Paths and Flow Matching

The discussion above clarifies a key separation: ODE dynamics provides a *mechanism* to transport distributions, but it does not tell us *what trajectory in distribution space* the transport should follow. In other words, the vector field u is a generative model only after we specify the intended family of intermediate marginals $\{P_t\}$ linking P_0 to P_{data}.

Flow matching makes this choice explicit by first prescribing a probability path $\{P_t\}_{t\in[0,1]}$ and then learning a velocity field whose induced marginals match that path. This leads to two intertwined questions: (i) how to design probability paths that are both expressive and tractable, and (ii) how to obtain supervised targets for $u(x,t)$ without simulating model trajectories. We address these questions next.

2.1 Probability Path Construction

A probability path $\{P_t\}_{t\in[0,1]}$ should satisfy

$$P_0 \text{ is an easy-to-sample base distribution} \quad \text{and} \quad P_1 = P_{\text{data}}.$$

At an abstract level, specifying such a path means choosing how probability mass should move from the base distribution toward the data distribution over time.

A direct specification of the full marginal family $\{P_t\}$ is often difficult. In practice, flow matching typically constructs P_t through a simpler *conditional probability path*. Let $X_1 \sim P_{\text{data}}$ be a data sample. We define a time-indexed family of conditional distributions

$$P_t(\cdot \mid X_1), \qquad t \in [0,1].$$

Intuitively, $P_t(\cdot \mid X_1)$ describes how an intermediate sample at time t should be distributed if its terminal destination is the data point X_1. The unconditional marginal path is then obtained by marginalizing over the data distribution:

$$P_t(A) = \int P_t(A \mid x_1) \mathrm{d}P_{\text{data}}(x_1) \quad \text{for any measurable set } A \subseteq \mathbb{R}^d. \tag{2.1}$$

If densities exist, we also have

$$p_t(x) = \int p_t(x \mid x_1) p_{\text{data}}(x_1) \mathrm{d}x_1.$$

This conditional construction is attractive because it is often much easier to specify a path toward an endpoint X_1 than to directly describe the full marginal law P_t. Such choices lead to simple closed-form expressions for the corresponding velocity targets, which is a main reason flow matching is computationally convenient.

Remark 2.1 (Generic conditional path)**.** The conditioning variable in a conditional probability path need not be the terminal data point X_1. More generally, one may introduce an auxiliary random variable $Z \sim P_Z$ and define a family of conditional distributions

$$P_t(\cdot \mid Z), \qquad t \in [0,1].$$

The corresponding marginal path is then obtained by averaging over the law of Z:

$$P_t(A) = \int P_t(A \mid z)\, \mathrm{d}P_Z(z) \quad \text{for any measurable set } A \subseteq \mathbb{R}^d.$$

If densities exist, this becomes

$$p_t(x) = \int p_t(x \mid z)\, p_Z(z)\, \mathrm{d}z.$$

The choice of Z depends on the path construction. For example (see also Figure 1 for a demonstration),

- if $Z = X_1$, then the path is conditioned on the terminal data point;
- if $Z = X_0$, then the path is conditioned on the initial base sample;
- if $Z = (X_0, X_1)$, then the path is conditioned on both endpoints.

Conditioning on both endpoints is particularly natural in interpolation-based constructions, where one explicitly specifies how mass moves from an initial sample X_0 to a terminal sample X_1.

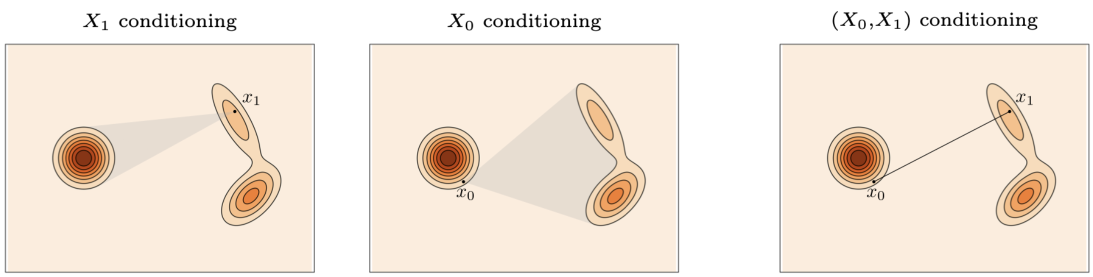

Figure 1: Different conditional probability path constructions [2].

Therefore, the essential idea is not the specific choice of X_1, but the use of a conditional variable Z that makes the path construction and the associated velocity field analytically tractable.

2.2 Velocity Estimation via Conditional Paths

Once a probability path $\{P_t\}_{t\in[0,1]}$ is chosen, the next task is to learn a velocity field whose flow realizes this path. At the population level, this means finding a vector field $u(x, t)$ such that the associated marginal density p_t satisfies the continuity equation

$$\frac{\partial}{\partial t} p_t(x) + \nabla \cdot \big(p_t(x) u(x,t)\big) = 0.$$

If such a vector field is learned, then solving the ODE

$$\frac{\mathrm{d}}{\mathrm{d}t} X_t = u(X_t, t)$$

transports the base distribution P_0 along the prescribed path and reaches the data distribution at time $t = 1$.

However, learning $u(\cdot, t)$ directly from the marginal path is difficult. Even if the family $\{P_t\}_{t=0}^1$ is conceptually specified, the corresponding marginal velocity field is usually not available in closed form. The key idea of flow matching is that the conditional path construction provides a tractable surrogate learning target.

We consider X_1 conditioning for simplicity. The argument applies to any Z conditioning. Suppose for each endpoint x_1, the conditional path $P_t(\cdot \mid X_1)$ is associated with a conditional velocity field $u(\cdot, t \mid X_1)$ satisfying

$$\frac{\partial}{\partial t} p_t(x \mid X_1 = x_1) + \nabla \cdot \big(p_t(x \mid X_1 = x_1) u(x, t \mid X_1 = x_1)\big) = 0.$$

Then the marginal velocity field is given by the conditional expectation

$$u(x, t) = \mathbb{E}\big[u(x, t \mid X_1) \mid X_t = x\big]. \tag{2.2}$$

In words, the marginal velocity at location x and time t is the average of the conditional velocities over all possible endpoints X_1, weighted by their posterior likelihood given the current state $X_t = x$.

Identity (2.2) is fundamental. It shows that the marginal velocity field can be recovered from conditional velocity fields, even though the posterior expectation itself may still be intractable. More importantly, it leads to a supervised regression formulation for learning the velocity field. Specifically, we parametrize the velocity field u by a neural network u_θ, then an ideal learning objective is

$$\mathcal{L}_{\mathrm{FM}}(\theta) = \int_0^1 \mathbb{E}_{X_t \sim P_t}\Big[\|u_\theta(X_t, t) - u(X_t, t)\|_2^2\Big] \mathrm{d}t. \tag{2.3}$$

Unfortunately, optimizing (2.3) encounters the same issue in score matching for diffusion models—ground-truth velocity u is unknown. To address this difficulty, flow matching leverages (2.2) and replaces the inaccessible marginal velocity $u(X_t, t)$ by the conditional velocity $u(X_t, t \mid X_1)$. We define the *conditional flow matching* objective as

$$\mathcal{L}_{\mathrm{CFM}}(\theta) = \int_0^1 \mathbb{E}_{X_1 \sim P_{\mathrm{data}}} \mathbb{E}_{X_t \sim P_t(\cdot \mid X_1)}\Big[\|u_\theta(X_t, t) - u(X_t, t \mid X_1)\|_2^2\Big] \mathrm{d}t. \tag{2.4}$$

The key advantage of (2.4) is that, for many conditional path constructions, the target $u(X_t, t \mid X_1)$ admits a simple analytic expression. As a result, training reduces to a standard supervised regression problem.

More surprisingly yet essential is that (2.4) is equivalent to the ideal objective (2.3) up to an additive constant. To see this, we fix any $t \in [0, 1]$ and expand the square

$$\begin{aligned}
&\mathbb{E}_{X_1 \sim P_{\mathrm{data}}} \mathbb{E}_{X_t \sim P_t(\cdot \mid X_1)}\Big[\|u_\theta(X_t, t) - u(X_t, t \mid X_1)\|_2^2\Big] \\
&= \mathbb{E}_{X_1 \sim P_{\mathrm{data}}} \mathbb{E}_{X_t \sim P_t(\cdot \mid X_1)}\Big[\|u_\theta(X_t, t) - u(X_t, t)\|_2^2\Big] \\
&\quad + \mathbb{E}_{X_1 \sim P_{\mathrm{data}}} \mathbb{E}_{X_t \sim P_t(\cdot \mid X_1)}\Big[\|u(X_t, t) - u(X_t, t \mid X_1)\|_2^2\Big] \\
&\quad + 2\mathbb{E}_{X_1 \sim P_{\mathrm{data}}} \mathbb{E}_{X_t \sim P_t(\cdot \mid X_1)}\Big[\big(u_\theta(X_t, t) - u(X_t, t)\big)^\top \big(u(X_t, t) - u(X_t, t \mid X_1)\big)\Big].
\end{aligned}$$

We now show that the cross term vanishes. Conditioned on X_t, the factor $u_\theta(X_t, t) - u(X_t, t)$ is deterministic. Therefore, we have

$$\begin{aligned}
&\mathbb{E}_{X_1 \sim P_{\mathrm{data}}} \mathbb{E}_{X_t \sim P_t(\cdot \mid X_1)}\Big[\big(u_\theta(X_t, t) - u(X_t, t)\big)^\top \big(u(X_t, t) - u(X_t, t \mid X_1)\big)\Big] \\
&= \mathbb{E}_{(X_1, X_t)}\Big[\big(u_\theta(X_t, t) - u(X_t, t)\big)^\top \big(u(X_t, t) - u(X_t, t \mid X_1)\big)\Big] \\
&= \mathbb{E}_{(X_1, X_t)}\Big[\big(u_\theta(X_t, t) - u(X_t, t)\big)^\top \mathbb{E}\Big[u(X_t, t) - u(X_t, t \mid X_1) \mid X_t\Big]\Big].
\end{aligned}$$

By (2.2), we have

$$\mathbb{E}\big[u(X_t, t \mid X_1) \mid X_t\big] = u(X_t, t),$$

and hence

$$\mathbb{E}\Big[u(X_t, t) - u(X_t, t \mid X_1) \,\Big|\, X_t\Big] = 0.$$

As a result, the cross term is zero, and we deduce

$$\mathcal{L}_{\mathrm{CFM}}(\theta) = \mathcal{L}_{\mathrm{FM}}(\theta) + \int_0^1 \mathbb{E}_{X_1 \sim P_{\mathrm{data}}} \mathbb{E}_{X_t \sim P_t(\cdot \mid X_1)} \Big[\|u(X_t, t) - u(X_t, t \mid X_1)\|_2^2\Big] \mathrm{d}t. \tag{2.5}$$

The second term in the right-hand side of (2.5) does not depend on θ. Consequently, minimizing the conditional flow matching objective (2.4) is equivalent to minimizing the ideal flow matching objective (2.3).

This equivalence is the central mathematical justification of flow matching. Although the true marginal velocity field $u(x, t)$ is inaccessible, it is enough to regress onto the conditional velocity field $u(x, t \mid X_1)$. The learned minimizer still recovers the same optimal marginal transport field.

Remark 2.2 (Training under Bregman divergence)**.** The conditional flow matching objective can be generalized beyond the squared Euclidean loss. Let $\phi : \mathbb{R}^d \to \mathbb{R}$ be a differentiable strictly convex function. The associated *Bregman divergence* is

$$D_\phi(y, z) = \phi(y) - \phi(z) - \langle \nabla\phi(z),\, y - z\rangle, \qquad y, z \in \mathbb{R}^d.$$

Replacing the squared loss in (2.4), we may define the conditional training objective

$$\mathcal{L}^{\phi}_{\mathrm{CFM}}(\theta) = \int_0^1 \mathbb{E}_{X_1 \sim P_{\mathrm{data}}} \mathbb{E}_{X_t \sim P_t(\cdot \mid X_1)} \Big[D_\phi\big(u(X_t, t \mid X_1),\, u_\theta(X_t, t)\big)\Big] \mathrm{d}t. \tag{2.6}$$

Similarly, the corresponding ideal marginal objective is

$$\mathcal{L}^{\phi}_{\mathrm{FM}}(\theta) = \int_0^1 \mathbb{E}_{X_t \sim P_t} \Big[D_\phi\big(u(X_t, t),\, u_\theta(X_t, t)\big)\Big] \mathrm{d}t. \tag{2.7}$$

The same conditional-expectation argument shows that the population minimizer of (2.6) and (2.7) coincide (verification is left as an exercise). The squared Euclidean loss is recovered by choosing

$$\phi(v) = \|v\|_2^2,$$

for which

$$D_\phi(y, z) = \|y - z\|_2^2.$$

Thus conditional flow matching can be viewed more generally as learning the marginal velocity field under a broad class of proper convex losses, with the squared loss being the most common special case.

3 Example: Affine Probability Paths

The previous section developed the general flow matching principle in an abstract form. We now study an important concrete family of probability paths, namely *affine probability paths* with Gaussian base distribution. This family is widely used in practice because the conditional velocity field can be written in closed form, which makes the conditional flow matching objective directly tractable.

3.1 Affine Conditional Paths with Gaussian Base Distribution

Let $\epsilon \sim \mathsf{N}(0, I)$ and data $X_1 \sim P_{\text{data}}$. Conditioned on the endpoint X_1, for two time scaling function α_t and β_t, we define the conditional path

$$X_t = \alpha_t X_1 + \beta_t \epsilon, \qquad t \in [0, 1]. \tag{3.1}$$

Equivalently, the associated conditional probability path is

$$P_t(\cdot \mid X_1) = \mathrm{Law}(\alpha_t X_1 + \beta_t \epsilon).$$

Since ϵ is Gaussian, we have

$$P_t(\cdot \mid X_1) = \mathsf{N}(\alpha_t X_1, \beta_t^2 I).$$

Averaging over $X_1 \sim P_{\text{data}}$ yields the marginal probability path

$$P_t(A) = \int P_t(A \mid x_1) \mathrm{d}P_{\text{data}}(x_1).$$

To ensure that the path interpolates from Gaussian noise to the data distribution, we impose the boundary conditions

$$(\alpha_0 = 0, \quad \beta_0 = 1) \quad \text{and} \quad (\alpha_1 = 1, \quad \beta_1 = 0).$$

Under these conditions,

$$X_0 = \epsilon \sim \mathsf{N}(0, I) \quad \text{and} \quad X_1 = X_1 \sim P_{\text{data}}.$$

Thus the path starts from a tractable Gaussian base distribution and ends at the data distribution.

3.2 Conditional Velocity Field

Because the path (3.1) is explicitly parameterized by time, we may differentiate it to obtain

$$\frac{\mathrm{d}}{\mathrm{d}t} X_t = \dot{\alpha}_t X_1 + \dot{\beta}_t \epsilon. \tag{3.2}$$

To express this velocity in terms of the current state X_t and the conditioned endpoint X_1, we have

$$\epsilon = \frac{X_t - \alpha_t X_1}{\beta_t},$$

whenever $\beta_t \neq 0$. Substituting this into (3.2) yields the conditional velocity field

$$u(X_t, t \mid X_1) = \dot{\alpha}_t X_1 + \frac{\dot{\beta}_t}{\beta_t}(X_t - \alpha_t X_1).$$

Equivalently, we obtain

$$u(X_t, t \mid X_1) = \frac{\dot{\beta}_t}{\beta_t} X_t + \left(\dot{\alpha}_t - \alpha_t \frac{\dot{\beta}_t}{\beta_t} \right) X_1. \tag{3.3}$$

This formula gives an explicit supervised target for conditional flow matching. Yet, the marginalized velocity is still hard to find explicitly.

3.3 Conditional Flow Matching Objective

Using the affine Gaussian path, the conditional flow matching objective becomes

$$\mathcal{L}_{\mathrm{CFM}}(\theta) = \int_0^1 \mathbb{E}_{X_1 \sim P_{\mathrm{data}}} \mathbb{E}_{\epsilon \sim \mathsf{N}(0,I)} \Big[\| u_\theta(\alpha_t X_1 + \beta_t \epsilon, t) - u(\alpha_t X_1 + \beta_t \epsilon, t \mid X_1) \|_2^2 \Big] \,\mathrm{d}t, \tag{3.4}$$

where we have substituted $X_t = \alpha_t X_1 + \beta_t \epsilon$ into (3.4). Plugging (3.3) into (3.4), we obtain

$$\mathcal{L}_{\mathrm{CFM}}(\theta) = \int_0^1 \mathbb{E}_{X_1 \sim P_{\mathrm{data}}} \mathbb{E}_{\epsilon \sim \mathsf{N}(0,I)} \left[\left\| u_\theta(\alpha_t X_1 + \beta_t \epsilon, t) - \left(\dot{\alpha}_t X_1 + \dot{\beta}_t \epsilon \right) \right\|_2^2 \right] \mathrm{d}t.$$

The form above is often the most convenient for implementation, since both the input

$$X_t = \alpha_t X_1 + \beta_t \epsilon$$

and the conditional velocity

$$\dot{\alpha}_t X_1 + \dot{\beta}_t \epsilon$$

are easy to simulate.

3.4 Choices of (α_t, β_t)

Several important choices of (α_t, β_t) fit into the flow matching framework.

Linear interpolation path. The simplest choice is

$$\alpha_t = t, \qquad \beta_t = 1 - t,$$

which gives

$$X_t = tX_1 + (1-t)\epsilon.$$

In this case, we have

$$\dot{\alpha}_t = 1, \qquad \dot{\beta}_t = -1,$$

and hence the conditional velocity is

$$u(x, t \mid x_1) = x_1 - \epsilon.$$

After eliminating ϵ in favor of x, it holds that

$$u(x, t \mid x_1) = x_1 - \frac{x - tx_1}{1-t} = \frac{x_1 - x}{1-t}.$$

Variance-preserving path. A diffusion-style choice is

$$X_t = \alpha_t X_1 + \beta_t \epsilon \quad \text{with} \quad \alpha_t^2 + \beta_t^2 = 1,$$

This is exactly the Gaussian noise corruption that appears in continuous-time diffusion models.

Remark 3.1 (Beyond affine Gaussian paths)**.** The affine Gaussian path is arguably the most important example in flow matching because it yields an explicit conditional velocity field and connects naturally to diffusion models. However, it is only one possible choice of probability path.

More generally, one may construct conditional paths in many other ways, as long as the resulting conditional velocity field remains tractable or can be well approximated. For example, one may consider

- *nonlinear interpolation paths*, where X_t depends on $Z = (X_1, \epsilon)$ through a nonlinear transformation [1];
- *optimal-transport-inspired paths*, which aim to move mass along straighter trajectories in distribution space [2];
- *rectified-flow paths*, which are designed to learn increasingly straight transports and enable fast ODE sampling [3].

Despite different choices on the probability path, the essential principle of flow matching remains the same: first choose a probability path connecting a simple base distribution to the data distribution, and then learn a velocity field or transport rule that realizes this path.

References

[1] Michael Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. *Journal of Machine Learning Research*, 26(209):1–80, 2025.

[2] Yaron Lipman, Marton Havasi, Peter Holderrieth, Neta Shaul, Matt Le, Brian Karrer, Ricky TQ Chen, David Lopez-Paz, Heli Ben-Hamu, and Itai Gat. Flow matching guide and code. *arXiv preprint arXiv:2412.06264*, 2024.

[3] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.