A FRIENDLY INTRODUCTION TO TRIANGULAR TRANSPORT

Maximilian Ramgraber
Delft University of Technology
Delft, Netherlands
m.ramgraber@tudelft.nl

Daniel Sharp
Massachusetts Institute of Technology
Cambridge, USA
dannys4@mit.edu

Mathieu Le Provost
Massachusetts Institute of Technology
Cambridge, USA
mleprovo@mit.edu

Youssef Marzouk
Massachusetts Institute of Technology
Cambridge, USA
ymarz@mit.edu

March 28, 2025

ABSTRACT

Decision making under uncertainty is a cross-cutting challenge in science and engineering. Most approaches to this challenge employ probabilistic representations of uncertainty. In complicated systems accessible only via data or black-box models, however, these representations are rarely known. We discuss how to characterize and manipulate such representations using *triangular transport maps*, which approximate any complex probability distribution as a transformation of a simple, well-understood distribution. The particular structure of triangular transport guarantees many desirable mathematical and computational properties that translate well into solving practical problems. Triangular maps are actively used for density estimation, (conditional) generative modelling, Bayesian inference, data assimilation, optimal experimental design, and related tasks. While there is ample literature on the development and theory of triangular transport methods, this manuscript provides a detailed introduction for scientists interested in employing measure transport without assuming a formal mathematical background. We build intuition for the key foundations of triangular transport, discuss many aspects of its practical implementation, and outline the frontiers of this field.

1 Motivation

Who is this tutorial for? This manuscript is an accessible introduction to *triangular transport*, a powerful and versatile method for generative modelling and Bayesian inference. In particular, triangular transport underpins effective algorithms for data assimilation, solving inverse problems, and performing simulation-based inference, with applications across myriad scientific disciplines.

This tutorial targets researchers with an interest in applied statistical methods but without a formal background in mathematics. Consequently, we will focus more on intuition, general concepts, and implementation, referring the reader to other relevant articles for more formal exposition and theory.

How does triangular transport work? Like other measure transport methods, triangular (measure) transport is a framework to transform one probability distribution into another. This operation is highly useful, as it allows us to characterize a complex target distribution π by transforming a simpler, known reference distribution η. Throughout this manuscript, we use orange to denote variables associated with the (problem-specific) target distribution and green to denote variables associated with the (user-defined) reference distribution, e.g., a standard Gaussian. The idea of coupling two distributions is used in a wide range of applications. In *generative modelling*, for example, we are usually interested in creating samples of a target distribution π, such as the distribution of 400×400 pixel images of cats. Measure transport methods approach this challenge by first learning a transport map $\mathbf{S}$ that transforms η to π, and then use $\mathbf{S}$ to convert reference samples $\mathbf{z} \sim \eta$ (here: 400×400 pixel images of Gaussian white noise) into samples from the target distribution $\mathbf{x} = \mathbf{S}(\mathbf{z}) \sim \pi$ (here: 400×400 pixel images of cats).

Some of these methods – among them triangular transport – can also characterize *conditionals* $\pi(\boldsymbol{a}|\boldsymbol{b}^*)$ of the *joint* target distribution $\pi(\boldsymbol{a}, \boldsymbol{b})$ of two random variables $\boldsymbol{a}$ and $\boldsymbol{b}$. Here blue denotes the fact that $\boldsymbol{b}^*$ is a deterministic, fixed value. Conditioning operations often arise as stochastic generalizations of evaluating deterministic processes (see Figure 1). As we will describe in Section 2, conditioning is also central to the Bayesian approach to statistical inference, which is an important tool across many scientific disciplines. We distinguish here between generating from $\pi(\boldsymbol{a}|\boldsymbol{b}^*)$ ("generate an image of a grey cat") and $\pi(\boldsymbol{a}, \boldsymbol{b})$ ("generate a color and a cat of that color") by assuming that $\boldsymbol{b}^*$ is determined outside of our control, either by user or application.

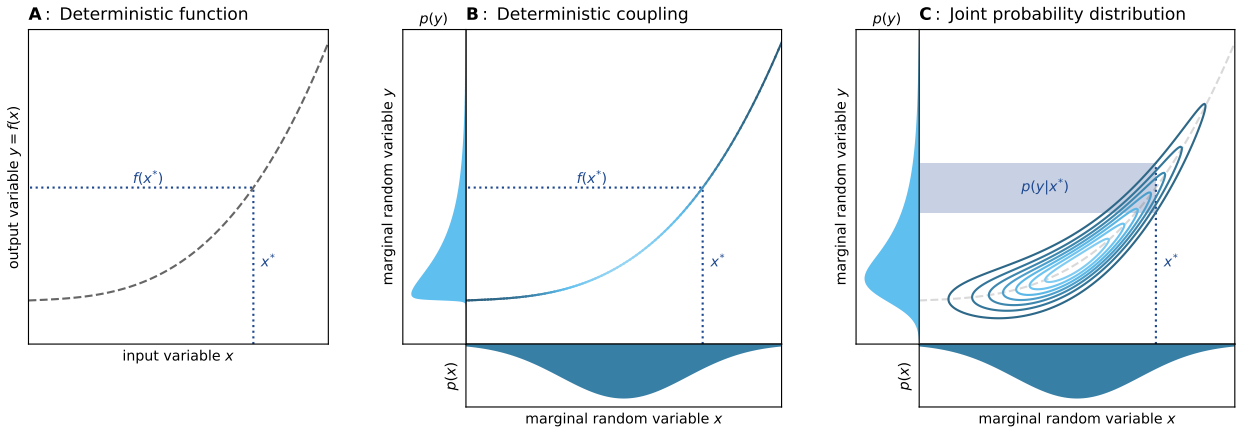

Figure 1: Progression from a fully deterministic to a fully stochastic system. (A) Numerical models are usually represented as deterministic functions. (B) In the presence of input uncertainty, deterministic functions encode a deterministic coupling which yields uncertain output (see Section 2.2). (C) If the function itself is uncertain, this coupling "blurs" into a joint probability distribution. Function evaluation now corresponds to characterizing a conditional distribution. Mind that (A) and (B) can also be parsed as degenerate joint probability distributions.

What makes triangular transport special? At the heart of triangular transport is their eponymous *triangular* structure. This structure sets them apart from other measure transport methods such as normalizing flows (e.g., Rezende and Mohamed, 2015; Kobyzev et al., 2021), which compose together many simpler but somewhat ad hoc transformations, often interleaved with permutations, and even from conditional normalizing flows (e.g., Van Den Oord et al., 2016), which parameterize normalizing flows in order to represent block triangular (rather than strictly triangular) maps. Triangular structure – discussed in greater detail in subsequent sections – has many important practical properties:

- **Parsimony**: The parameterization of the map function is at the user's discretion (see Section 3.1). This means we can adjust the map's overall complexity, down to the complexity with which it resolves individual variables and variable dependencies. This allows us to implement nonlinear maps that are as complex as necessary, and yet as simple as possible (see Section 4.2).
- **Sparsity**: Triangular maps have a natural ability to exploit *conditional independence*. This improves their computational efficiency, which enables these maps to scale to high-dimensional settings. Further, such structure makes them highly robust to spurious correlations and smaller sample sizes (see Section 2.3.3).
- **Numerical convenience**: Constructing triangular maps boils down to parameterizing simple one-dimensional monotone functions, a task with a rich body of supporting literature. Because of this, these maps are easy to optimize and invert, which we investigate in detail.
- **Explainability**: Triangular maps have a clear correspondence between their constituent elements and the statistical features they represent (see Section 2.3). We can then readily describe different factorizations of the target distribution using the elements of a triangular map, with a particular focus on combinations of various conditional and marginals of the target.

In what applications has triangular transport been successful? Triangular transport has been applied to a wide range of statistical problems in many different disciplines. Examples of such applications include:

- **Bayesian inference**: Triangular transport lends itself exceptionally well to the sampling of conditional distributions. As such, it has found application in both variational (El Moselhy and Marzouk, 2012) and simulation-based inference (Marzouk et al., 2017; Rubio et al., 2023; Baptista et al., 2024a), for large-scale inverse problems (Brennan et al., 2020) and in applications with multiscale structure (Parno et al., 2016).
- **Data assimilation**: Triangular transport provides true nonlinear generalizations of popular filtering (Spantini et al., 2022) and smoothing (Ramgraber et al., 2023a,b) algorithms such as the ensemble Kalman filter and smoother, and their many variants (Grange et al., 2024).
- **Density estimation and generative modelling**: The coupling learned by triangular transport is highly useful for the estimation (Wang and Marzouk, 2022; Martinez-Sanchez et al., 2024; López-Marrero et al., 2024) and sampling (Irons et al., 2022) of non-Gaussian probability distributions, even in high dimensions (Katzfuss and Schäfer, 2024).
- **Optimal experimental design**: Due to the close connections between conditional densities and expected information gain or mutual information, triangular transport maps are useful for estimating common objectives in Bayesian optimal experimental design (Huan et al., 2024; Koval et al., 2024; Li et al., 2024).

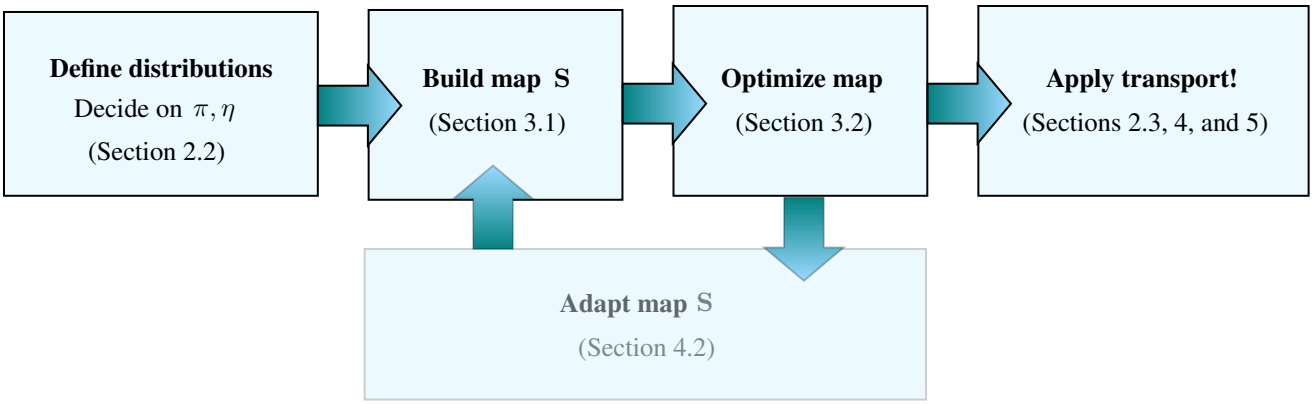

Figure 2: Flowchart of the main steps required to define, build, optimize, and apply triangular maps, with links to the relevant sections.

Further applications of triangular transport include methods for joint state-parameter inference in state-space models (Spantini et al., 2018; Grashorn et al., 2024; Zhao and Cui, 2024), solving Fokker–Planck equations (Zeng et al., 2023), stochastic programming (Backhoff et al., 2017), and even the discovery of causal models from data (Akbari et al., 2023; Xi et al., 2023).

How is this tutorial structured? In the following, we will provide an intuition-focused introduction to the theoretical basics of triangular transport (Section 2), discuss practical aspects related to their implementation in code (Section 3), and conclude with a brief overview of interesting research directions (Section 5). For researchers interested chiefly in practical implementation, a flow chart of the most important steps in the construction and application of triangular transport is provided in Figure 2. First, we define the target π and reference η (Section 2.3). Then, we structure, parameterize (Section 3.1), and optimize (Section 3.2) the triangular map. Finally, we can deploy the map in the application of our choice, with some practical heuristics listed in Section 4. The tutorial will involve several recurring variables, which are summarized in Table 1.

2 Theory

2.1 Bayesian inference

To begin, let us briefly revisit some basic concepts of Bayesian inference which will serve to motivate the operations explored in the following sections. In short, Bayesian inference is based on *Bayes' theorem*: given two random variables (RVs) $\mathbf{a}$, $\mathbf{b}$ with joint probability density function (pdf) $p(\mathbf{a}, \mathbf{b})$, we see

$$p\left(\mathbf{a}|\mathbf{b}^*\right) = \frac{p\left(\mathbf{a}\right) p\left(\mathbf{b}^*|\mathbf{a}\right)}{p\left(\mathbf{b}^*\right)}, \tag{1}$$

Equation (1)subsumes three sequential operations (see Figure 3; e.g., Gelman et al., 2013):

1. First, the marginal prior $p\left(\mathbf{a}\right)$ is combined with a conditional observation model (sometimes also called *likelihood model*) $p\left(\mathbf{b}|\mathbf{a}\right)$, yielding a joint probability distribution $p\left(\mathbf{a}, \mathbf{b}\right) = p\left(\mathbf{a}\right) p\left(\mathbf{b}|\mathbf{a}\right)$ over all possible

Table 1: Recurring notation and variables.

bold font	vector-valued variable or function	Roman font	scalar-valued variable or function
$\mathbf{S}$	Target-to-reference map	S_k	k-th map component function
$\mathbf{R}$	Reference-to-target map (see Section 3.2.2)	τ	twice Archimedes' constant, i.e., $6.283185...$
π	target distribution of interest	η	reference distribution, often standard Gaussian
orange	variable associated with π	green	variable associated with η
$\mathbf{x} \sim \pi$	target random variable	$\mathbf{z} \sim \eta$	reference random variable
$\mathbf{X}^i$	ith realization of $\mathbf{x} \sim \pi$	$\mathbf{Z}^i$	$i$the realization of $\mathbf{z} \sim \eta$
$S^\sharp \eta$	pullback distribution	$S_\sharp \pi$	pushforward distribution
K	number of target dimensions	N	ensemble size
p	generic probability density function (pdf)	$\mathbf{a}, \mathbf{b}$	generic random variables
$\mathbf{y}^*$	conditioning variable	$\mathbf{x}^*$	conditioned variable $\mathbf{x}^* \sim p(\mathbf{x}\|\mathbf{y}^*)$
c	basis function coefficient	r	rectifier ($r : \mathbb{R} \to \mathbb{R}^+$)
f	monotone function	g	nonmonotone function

combinations of $\mathbf{a}$ and $\mathbf{b}$. This joint distribution describes how the variable of interest $\mathbf{a}$ and the predicted observations $\mathbf{b}$ relate to each other.

2. Next, this joint distribution is conditioned on a specific observation $\mathbf{b}^*$. In practice, this means evaluating $p(\mathbf{a}, \mathbf{b})$ for all possible values $\mathbf{a}$ while keeping $\mathbf{b}$ fixed at the value of $\mathbf{b}^*$. This extracts a slice $p(\mathbf{a}, \mathbf{b}^*)$ of this joint distribution at $\mathbf{b}^*$ along different values of $\mathbf{a}$.

3. Since the probability densities along this slice do not generally integrate to 1, this slice does not constitute a valid pdf. The final step thus normalizes the probability densities against the slice's probability mass $p(\mathbf{b}^*) = \int p(\mathbf{t}, \mathbf{b}^*)\, d\mathbf{t}$, yielding the posterior pdf $p(\mathbf{a}|\mathbf{b}^*)$.

In summary, Bayes' theorem first constructs a joint distribution $p(\mathbf{a}, \mathbf{b})$ from a prior $p(\mathbf{a})$ and an observation model $p(\mathbf{b}|\mathbf{a})$, then conditions it on a specific observation value $\mathbf{b}^*$ and re-normalizes. In consequence, one could reformulate Equation (1) equivalently as:

$$p(\mathbf{a}|\mathbf{b}^*) = \frac{p(\mathbf{a}, \mathbf{b}^*)}{\int p(\mathbf{t}, \mathbf{b}^*)\, d\mathbf{t}}, \tag{2}$$

where $p(\mathbf{a}, \mathbf{b}^*)$ evaluates the joint pdf $p(\mathbf{a}, \mathbf{b})$ for all possible $\mathbf{a}$ while keeping $\mathbf{b}$ fixed at $\mathbf{b}^*$, and the denominator acts as a normalizing constant. This equation, or reformulation thereof, lie at the heart of all Bayesian inference algorithms. Unfortunately, it is generally impossible to formulate $p(\mathbf{a}, \mathbf{b})$ in closed form, which in turn makes it difficult to evaluate the posterior $p(\mathbf{a}|\mathbf{b}^*)$. To overcome this challenge, different Bayesian inference methods use different strategies. As we shall see in the following, triangular transport solves this challenge by first approximating an almost arbitrary joint pdf $p(\mathbf{a}, \mathbf{b})$ by using the concept of measure transport (Section 2.2.1). Crucially, this transformation then allows us to evaluate any of its conditionals $p(\mathbf{a}|\mathbf{b}^*)$ (Section 2.3.2).

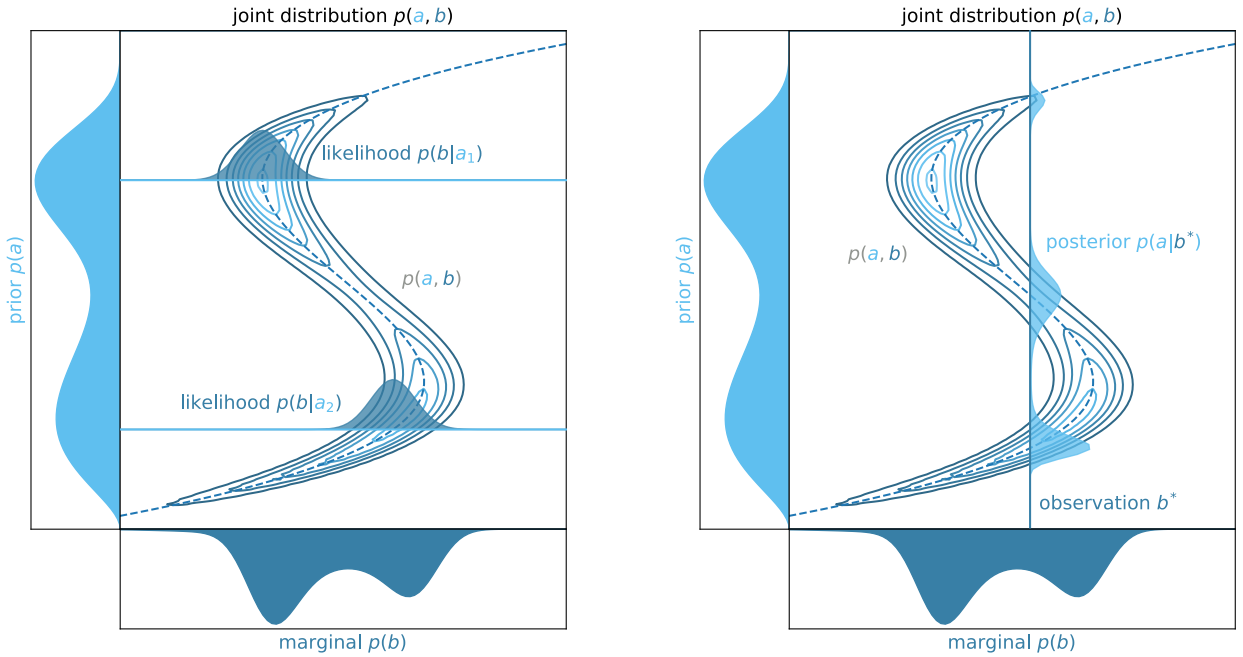

Figure 3: Schematic illustration of Bayes theorem, using a Beta mixture prior $p(a)$ and an observation model $p(b|a) = \mathcal{N}\left(\mu = 2a^3 - a, \sigma = 0.075\right)$. Left: We can create a joint distribution $p(a, b)$ from a prior $p(a)$ and the observation model $p(b|a)$. Right: conditioning this joint distribution on a specific value b^* retrieves a posterior $p(a|b^*)$.

The remainder of this tutorial drops this generic notation for two RVs $\mathbf{a}$ and $\mathbf{b}$ jointly distributed as $p(\mathbf{a}, \mathbf{b})$ in favour of a single RV $\mathbf{x}$ distributed according to a distribution π. You can think of $\mathbf{x}$ as an augmented RV $\mathbf{x} = [\mathbf{b}, \mathbf{a}]$, and of the joint density as $\pi(\mathbf{x}) = p(\mathbf{a}, \mathbf{b})$. Transport methods generally operate on this joint distribution $p(\mathbf{a}, \mathbf{b})$; we focus primarily on the ones that will allow us to characterize the desired conditionals $p(\mathbf{a}|\mathbf{b})$.

2.2 The change-of-variables formula

The key to understand transport methods is the *change-of-variables formula*. This formula allows us to relate a RV $\mathbf{x}$ associated with a complicated target pdf π, known only to proportionality or through samples, to a second RV $\mathbf{z}$, associated with a much simpler, user-specified reference pdf η through an invertible, differentiable transformation $\mathbf{S}$. In essence, the change-of-variables formula describes what happens to pdfs when they are subjected to specific transformations. For scalar-valued RVs x and z, the change-of-variables formula is defined as

$$\begin{aligned} \pi(x) &= S^\sharp \eta(x) = \eta(S(x)) \left| \frac{\partial S(x)}{\partial x} \right|, \\ \eta(z) &= S_\sharp \pi(z) = \pi(S^{-1}(z)) \left| \frac{\partial S^{-1}(z)}{\partial z} \right|, \end{aligned} \tag{3}$$

where $z = S(x)$ and $x = S^{-1}(z)$, and the *pullback density*[1] $S^\sharp \eta(x)$ obtains π by applying the inverse map S^{-1} to the reference distribution η. The alternate form in the second line of Equation (3) reflects the fact that since S is invertible,

[1]The *pullback* density $S^\sharp \eta(\mathbf{x})$ is the result of applying of the *inverse* map S^{-1} to a RV $z \sim \eta$.

each distribution π and η can be expressed in terms of the other through either the (forward) map S or its inverse S^{-1}. Consequently, the *pushforward pdf*[2] $S_\sharp\pi(z)$ likewise approximates η by applying the forward map S to the target[3] π. For multivariate RVs $\mathbf{x}$ and $\mathbf{z}$, the same principle applies. Given a multivariate monotone function $\mathbf{S}$, Equation (3) generalizes to:

$$\begin{aligned}\pi(\mathbf{x}) &= \mathbf{S}^\sharp\eta(\mathbf{x}) = \eta(\mathbf{S}(\mathbf{x}))\,|\det\nabla_{\mathbf{x}}\mathbf{S}(\mathbf{x})|\\ \eta(\mathbf{z}) &= \mathbf{S}_\sharp\pi(\mathbf{z}) = \pi(\mathbf{S}^{-1}(\mathbf{z}))\,\left|\det\nabla_{\mathbf{z}}\mathbf{S}^{-1}(\mathbf{z})\right|\end{aligned} \tag{4}$$

The change-of-variables formula in Equation (3) has a surprisingly intuitive interpretation. It states that the probability density $\eta(z)$ at a point of interest z after the transformation equals the original probability density $\pi(x)$ at the pre-transformation point $x = S^{-1}(z)$, adjusted for any deformation the transformation might have induced at this location $|\partial S(x)/\partial x|$. Equation (4) extends this notion to multi-dimensional systems. Intuitively, the absolute value of the determinant of the Jacobian of $\mathbf{S}$, namely $|\det\nabla_{\mathbf{x}}\mathbf{S}(\mathbf{x})|$, measures the inflation/deflation of an infinitesimal volume centered about $\mathbf{x}$ by the map $\mathbf{S}$. If $|\det\nabla_{\mathbf{x}}\mathbf{S}(\mathbf{x})| = 1$, the map $\mathbf{S}$ preserves infinitesimal volumes about $\mathbf{x}$. The compensation term, similar to 'u-substitution' in calculus, is necessary because transformations $\mathbf{S}$ "stretch" or "squeeze" the spaces they are applied to. Accounting for this spatial distortion ensures that the probability mass is preserved and thus the transformed distribution $\eta(\mathbf{z})$ remains a valid pdf.

The change-of-variables formula allows us to describe how a probability distribution π changes when subjected to a specific transformation $\mathbf{S}$. An example is provided in Figure 4, which shows how a non-Gaussian target pdf π can be related to a Gaussian reference pdf η through an invertible transformation; this invertibility is equivalent to monotonicity in the scalar case.

2.2.1 Connection to transport methods

The change-of-variables formula has three knobs to tune: the original distribution π, the map $\mathbf{S}$, and its transformed output η. When considering the change-of-variables in elementary calculus courses, π and $\mathbf{S}$ are often assumed known, and we seek its transformed output η. Transport methods choose a slightly different approach. We assume π and η to be known (at least partially), and instead seek the specific map $\mathbf{S}$ which relates the two distributions to each other.

As discussed in Section 1, we generally do not know the target distribution π in closed form. Often, the target density π is known only partially, either through samples[4] $\mathbf{X} \sim \pi$ or up to proportionality ($\tilde{\pi} = m\pi$, where $m > 0$ is an unknown constant)[5]. On the other hand, the reference distribution η is defined as a simple, well-known distribution, often a standard Gaussian pdf $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$ (Figure 5). The map $\mathbf{S}$ is identified by minimizing an objective function over a specified class of functions, see the discussion in Section 3.2. By finding this map, we learn how to construct the unknown target distribution π by transforming a well-defined reference distribution η.

[2]The *pushforward* density $S_\sharp\pi(\mathbf{z})$ is the result of applying the *forward* map S to a RV $x \sim \pi$.

[3]A mnemonic bridge to remember the $\sharp$ notation is that the pull*back* relies on the *inverse* map $\mathbf{S}^{-1}$ to sample, and has the $\sharp$ in the superscript where the -1 would be.

[4]Sample approximations are common if π is only known from a dataset or if the prior can be sampled but not evaluated.

[5]Unnormalized densities $\tilde{\pi}$ can arise in, e.g., Bayesian statistics, when the prior and likelihood can be evaluated at least point-wise, but it is infeasible to quantify the model evidence (see Equation (1)).

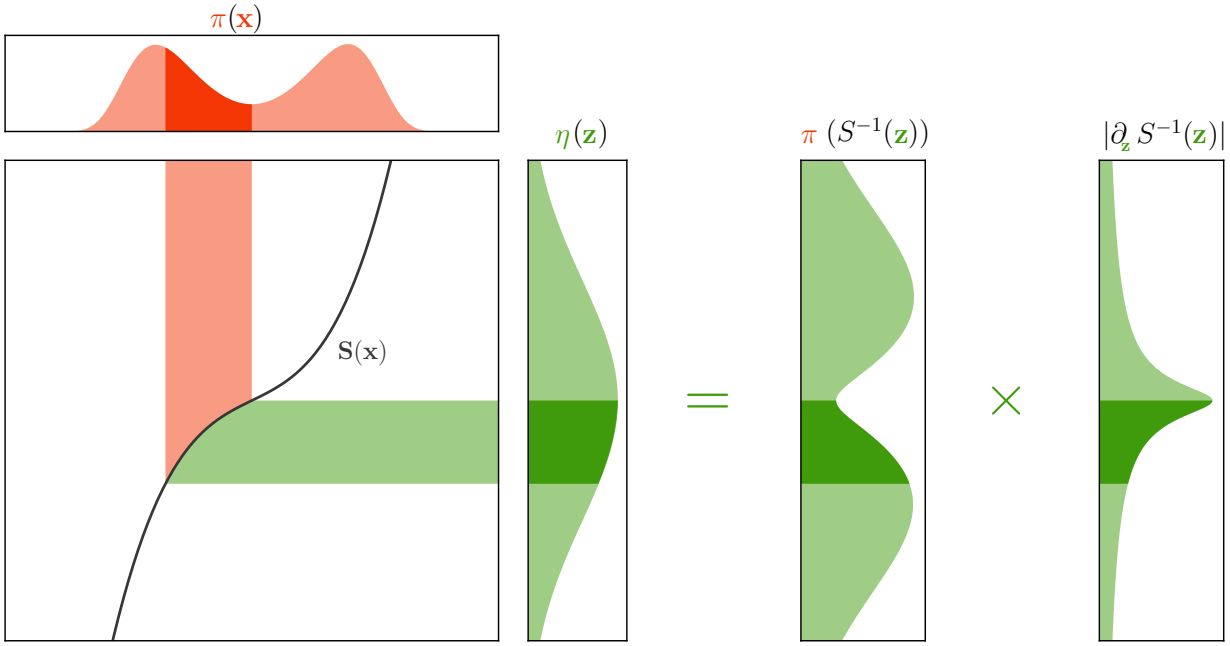

Figure 4: Illustration of the change-of-variables formula. A monotone function S allows us to relate a RV x associated with a pdf π to a RV z associated with a pdf η. We can evaluate $\eta(z)$ by evaluating the target π at the pre-image of z, that is to say $\pi(S^{-1}(z))$, then multiplying it with the absolute inverse map's gradient $|\partial_z S^{-1}(z)|$.

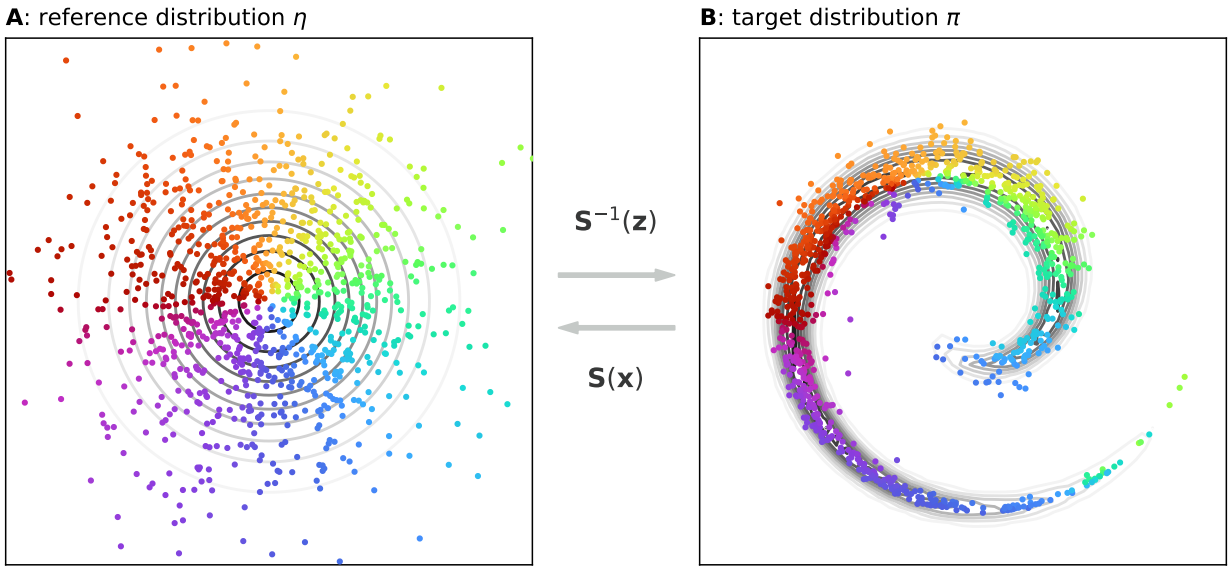

Figure 5: A transport map $\mathbf{S}$ relates a RV $\mathbf{x}$ associated with the target π to a RV $\mathbf{z}$ associated with the reference η. If the map is monotone, it can be applied both ways.

Among its other uses, learning the map $\mathbf{S}$ allows us to cheaply draw new samples from the target distribution π. This is achieved by first sampling the reference η, then applying the inverse map $\mathbf{S}^{-1}$ to the resulting reference samples $\mathbf{z}$. This is especially useful in applications where sampling the target conventionally would involve computationally expensive expensive simulations of, e.g., partial differential equations (*emulation*), or systems in which the sample-generating process is not known exactly (*generative modelling*: e.g., Baptista et al., 2024c).

2.3 Triangular maps and their uses

Many classes of functions are viable choices for the transport map $\mathbf{S}$. Common examples include normalizing flows and GANs. However, an especially useful class among these are *triangular* transport maps. In addition to sampling the target π, triangular maps are unique in allowing us to sample *conditionals* of π, which makes them a flexible and versatile tool for Bayesian inference. Triangular maps are structured as follows:

$$\mathbf{S}(\mathbf{x}) = \begin{bmatrix} S_1(x_1) \\ S_2(x_1, x_2) \\ \vdots \\ S_K(x_1, \dots, x_K) \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_K \end{bmatrix} = \mathbf{z}, \tag{5}$$

where the full map $\mathbf{S} : \mathbb{R}^K \to \mathbb{R}^K$ is comprised of map components $S_k : \mathbb{R}^k \to \mathbb{R}$, $k = 1, \dots, K$, each of which depends only on the first k entries of the target RV $\mathbf{x} = [x_1, \dots, x_K]^\top$ and we *enforce* that $\partial_{x_k} S_k(x_1, \dots, x_k) > 0$ for any feasible choice of $x_1, \dots, x_k$. When all of $S_1, \dots, S_K$ satisfy this, we call the map $\mathbf{S}$ "monotone". The eponymous *triangular* nature of $\mathbf{S}$ refers to the fact that the map's partial derivatives with regards to $\mathbf{x}$ are lower-triangular; that is to say, the Jacobian matrix $\nabla\mathbf{S}$ has all zeros above its diagonal. This structure—also known as a *Knothe–Rosenblatt rearrangement* (Knothe, 1957; Rosenblatt, 1952) or KR map—has a number of highly desirable properties:

1. Over all functions satisfying this structure, there is one that uniquely couples π and η under mild conditions (e.g., Marzouk et al., 2017).

2. This triangular structure allows us to evaluate the **determinant of the map's Jacobian** $\det \nabla\mathbf{S}(\mathbf{x})$ efficiently as the product of its diagonal entries (Marzouk et al., 2017), which proves highly useful for the map's optimization (see Section 3.2), not to mention when performing the density estimation itself for the changed variables:

$$\det \nabla\mathbf{S}(\mathbf{x}) = \prod_{k=1}^{K} \frac{\partial S_k(x_1, \dots, x_k)}{\partial x_k}. \tag{6}$$

3. Triangular maps are **easily invertible**. In particular, we pick a class of functions that are monotone in their last input x_k (with no requirements on the other inputs $x_1, \dots, x_{k-1}$), i.e. $\partial_{x_k} S_k > 0$ everywhere; this guarantees the monotonicity and thus eases our inversion computation. We will discuss ways to guarantee this property in Section 3.1.1.

4. Perhaps most importantly, triangular maps naturally **factorize the target distribution** into a product of marginal conditional pdfs. We will investigate this property in greater detail in the following sections.

2.3.1 Map inversion

In the forward map evaluation (Equation (5)), each of the map's constituent map component functions S_k can be evaluated independently, even in parallel, and then assembled into the full reference vector $\mathbf{z}$. However, the same does not hold for the inverse map:

$$\mathbf{S}^{-1}(\mathbf{z}) = \begin{bmatrix} S_1^{-1}(z_1) \\ S_2^{-1}(z_2; x_1) \\ S_3^{-1}(z_2; x_1, x_2) \\ \vdots \\ S_K^{-1}(z_K; x_1, \dots, x_{K-1}) \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_K \end{bmatrix} = \mathbf{x}. \tag{7}$$

Here, the inverse map's *component functions* S_k^{-1} must be evaluated in sequence and cannot be evaluated independently. This process begins by inverting the first map component $S_1^{-1}(z_1)$, a trivial one-dimensional root finding problem[6], which yields x_1. This output x_1 serves as auxiliary input for the second map component's inversion $S_2^{-1}(z_2; x_1)$, yielding another one-dimensional root-finding problem, which provides x_2. All subsequent map component inversions are similar one-dimensional root-finding problems that likewise depend on the outcomes of previous inversions. This dependence of each inversion S_k^{-1} on each of $x_1, \dots, x_{k-1}$, the outcomes of previous inversions, effectively factorizes the target distribution as a product of marginal conditionals (Villani, 2007):

$$\pi(\mathbf{x}) = \underbrace{\pi(x_1)}_{S_1^{-1}(z_1)} \underbrace{\pi(x_2|x_1)}_{S_2^{-1}(z_2;x_1)} \underbrace{\pi(x_3|x_1,x_2)}_{S_3^{-1}(z_3;x_1,x_2)} \dots \underbrace{\pi(x_K|x_1,\dots,x_{K-1})}_{S_K^{-1}(z_K;x_1,\dots,x_{K-1})}, \tag{8}$$

where each term corresponds to, and is in turn sampled by, one of the inverse map components indicated in the underbraces[7]. In other words, for a particular sample i, each row of Equation (7) can be used to generate a sample X_k^i from a particular marginal distribution conditioned on $\mathsf{X}_1, \dots, \mathsf{X}_{k-1}$:

$$\mathsf{X}_k^i = S_k^{-1}(\mathsf{Z}_k^i; \mathsf{X}_1^i, \dots, \mathsf{X}_{k-1}^i) \sim S_k^\sharp \eta_k = \pi(x_k|x_1, \dots, x_{k-1}), \tag{9}$$

where $S_k^\sharp \eta_k$ is the pullback of the one-dimensional marginal reference η_k.

2.3.2 Sampling conditionals

As it turns out, the factorization of the target distribution π in Equations (8) and (9) also allows us to sample *conditionals* of π, including the Bayesian posterior $p(\mathbf{a}|\mathbf{b})$ (assuming $p := \pi$ and $[\mathbf{b}, \mathbf{a}] := [\mathbf{x}_{1:k}, \mathbf{x}_{k+1:K}]$). This can be achieved by manipulating the inversion process. First, observe that the factorization in Equation (8) can be aggregated into two blocks:

[6] This root finding problem has a single unique solution within the domain of S_k^{-1}, since we require S_k be monotone in x_k.

[7] This is only the case if the reference distribution η has no dependence between any of its marginals (Spantini et al., 2022), i.e., $\eta(\mathbf{z}) = \eta_1(z_1)\eta_2(z_2)\dots\eta_K(z_K)$; the standard Gaussian fulfills this property.

$$\pi(\mathbf{x}) = \underbrace{\pi(\mathbf{x}_{1:k})}_{\mathbf{S}_{1:k}^{-1}(\mathbf{z}_{1:k})} \underbrace{\pi(\mathbf{x}_{k+1:K}|\mathbf{x}_{1:k})}_{\mathbf{S}_{k+1:K}^{-1}(\mathbf{z}_{k+1:K};\mathbf{x}_{1:k})}. \tag{10}$$

Similarly, we can aggregate the map component functions into two blocks:

$$\mathbf{S}^{-1}(\mathbf{z}) = \begin{bmatrix} \mathbf{S}_{1:k}^{-1}(\mathbf{z}_{1:k}) \\ \mathbf{S}_{k+1:K}^{-1}(\mathbf{z}_{k+1:K};\mathbf{x}_{1:k}) \end{bmatrix} = \begin{bmatrix} \mathbf{x}_{1:k} \\ \mathbf{x}_{k+1:K} \end{bmatrix} = \mathbf{x}. \tag{11}$$

Instead of evaluating of Equation (7) from top to bottom (S_1^{-1} to S_K^{-1}), if we are interested in sampling conditionals, we may skip the upper map block $\mathbf{S}_{1:k}^{-1}$ and replace its corresponding output $\mathbf{x}_{1:k} = [x_1, \dots, x_k]$ with arbitrary, user-specified values $\mathbf{x}_{1:k}^* = [x_1^*, \dots, x_k^*]$. This results in the following truncated inversion for the lower map block $\mathbf{S}_{k+1:K}^{-1}$,

$$\mathbf{S}_{k+1:K}^{-1}(\mathbf{z}_{k+1:K};\mathbf{x}_{1:k}^*) = \begin{bmatrix} S_{k+1}^{-1}(z_{k+1};\mathbf{x}_{1:k}^*) \\ S_{k+2}^{-1}(z_{k+2};\mathbf{x}_{1:k}^*, x_{k+1}^*) \\ \vdots \\ S_K^{-1}(z_K;\mathbf{x}_{1:k}^*, x_{k+1}^*, \dots, x_{K-1}^*) \end{bmatrix} = \begin{bmatrix} x_{k+1}^* \\ x_{k+2}^* \\ \vdots \\ x_K^* \end{bmatrix} = \mathbf{x}_{k+1:K}^*. \tag{12}$$

Resuming the inversion starting with the map component inverse S_{k+1}^{-1} thus yields samples $\mathbf{x}_{k+1:K}^*$ from the conditional $\pi(\mathbf{x}_{k+1:K}|\mathbf{x}_{1:k}^*)$ instead of the target $\pi(\mathbf{x})$. Equivalent to Equation (8), the corresponding conditional distribution now factorizes as:

$$\pi(\mathbf{x}_{k+1:K}|\mathbf{x}_{1:k}^*) = \underbrace{\pi(x_{k+1}|\mathbf{x}_{1:k}^*)}_{S_{k+1}^{-1}(z_{k+1};\mathbf{x}_{1:k}^*)} \underbrace{\pi(x_{k+2}|\mathbf{x}_{1:k}^*, x_{k+1})}_{S_{k+2}^{-1}(z_{k+2};\mathbf{x}_{1:k}^*, x_{k+1}^*)} \cdots \underbrace{\pi(x_K|\mathbf{x}_{1:k}^*, \mathbf{x}_{k+1:K-1})}_{S_K^{-1}(z_K;\mathbf{x}_{1:k}^*, x_{k+1}^*, \dots, x_{K-1}^*)}, \tag{13}$$

This means that the manipulated triangular map inversion in Equation (7) allows us to sample conditionals of the target distribution π for arbitrary $\mathbf{x}_{1:k}^*$, or to estimate the density of this conditional distribution. Recalling Section 2.1, it is plain to see why this operation proves extremely useful in Bayesian inference: If we define our target pdf π as the joint distribution $p(\mathbf{a}, \mathbf{b})$ (using the notation of Section 2.1) between the RV of interest $\mathbf{a}$ and the observation predictions $\mathbf{b}$, and consider the manipulated samples $\mathbf{x}_{1:k}^*$ to be the observations $\mathbf{b}^*$ of $\mathbf{b}$, then the manipulated inversion in Equation (12) samples the posterior $p(\mathbf{a}|\mathbf{b}^*)$. An illustration of the forward, inverse, and conditional mapping operations is provided in Figure 6.

2.3.3 Conditional independence

A related, very useful property of triangular transport maps is that they naturally allow for the exploitation of *conditional independence*[8] by construction. Recalling the map's generic factorization of the conditionals of target π in Equation (8),

[8] By default, many statistical methods assume that all RVs directly influence each other. *Conditional independence* arises whenever two RVs $\boldsymbol{a}$ and $\boldsymbol{c}$ only affect each other indirectly via a third RV $\boldsymbol{b}$. In this case, we say that $\boldsymbol{a}$ is conditionally independent of $\boldsymbol{c}$ (and vice versa) given $\boldsymbol{b}$, represented symbolically as $\boldsymbol{a} \perp\!\!\!\perp \boldsymbol{c} \mid \boldsymbol{b}$. As an example, consider a as the chance of rain, b as the chance of wet ground (which can, but does not have to be caused by rain), and c as the chance of slipping. As rain (a) only affects the chance of slipping (c) via wet ground (b), we have $a \perp\!\!\!\perp c \mid b$.

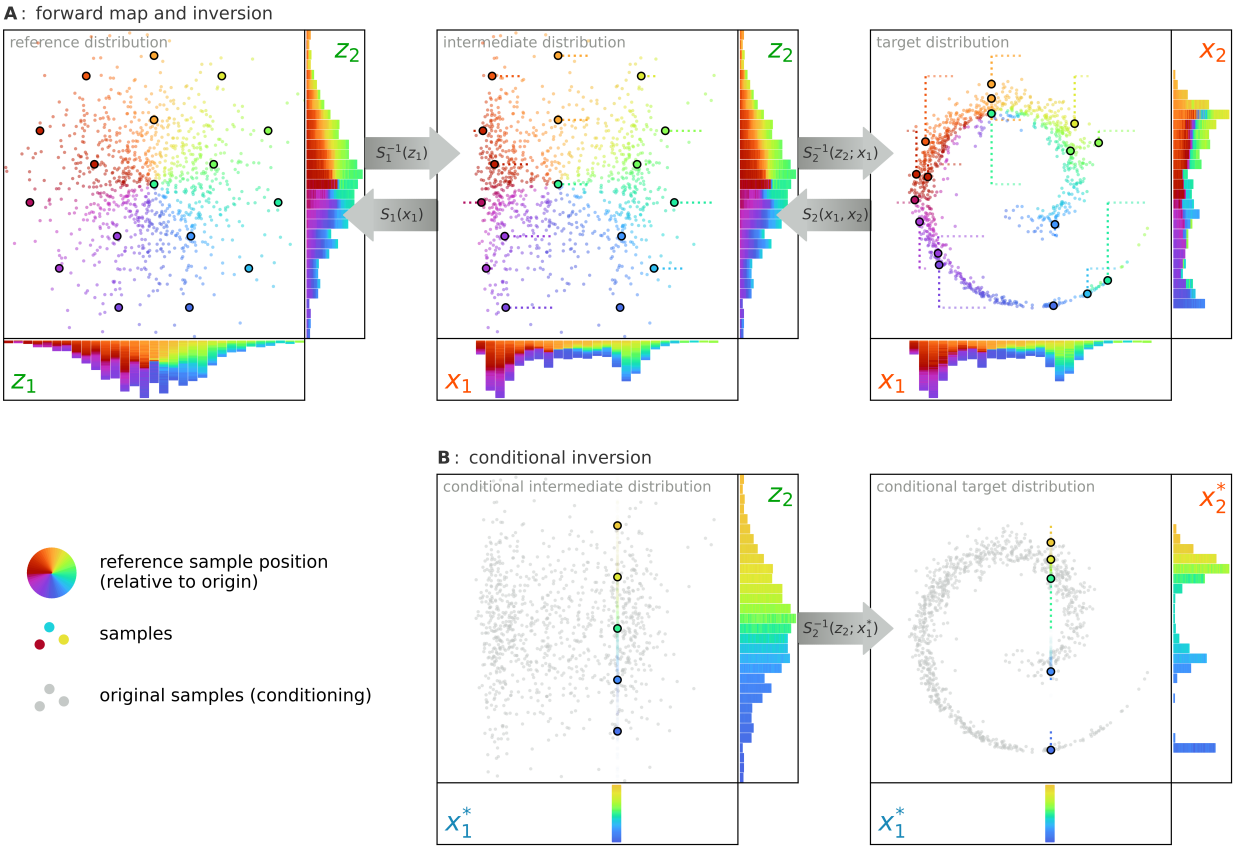

Figure 6: (A) The forward map (top row, from right) and its inverse (top row, from left) operate via implicit intermediate distributions (center), as the map components transform the distributions one marginal at a time. (B) Supplying the inverse map with a manipulated intermediate distribution (bottom row, center) as a starting point will instead sample conditionals of the target distribution.

we might ask ourselves what happens in systems in which we can exploit conditional independence. For example, if we have a target distribution $\pi(\mathbf{x}_{1:4})$ and conditional independence properties $x_3 \perp\!\!\!\perp x_1 | x_2$ and $x_4 \perp\!\!\!\perp x_1, x_2 | x_3$ (corresponding to *Markov structure*), the map's factorization could be reduced as follows:

$$\begin{aligned}\pi(\mathbf{x}_{1:4}) &= \pi(x_1)\,\pi(x_2|x_1)\,\pi(x_3|\cancel{x_1}, x_2)\,\pi(x_4|\cancel{x_1}, \cancel{x_2}, x_3) \\ &= \pi(x_1)\,\pi(x_2|x_1)\,\pi(x_3|x_2)\,\pi(x_4|x_3)\,.\end{aligned} \tag{14}$$

We refer to this reduction as *sparsification*. Triangular transport maps allows us to leverage these conditional independence properties by simply dropping the corresponding arguments from the map components S_k:

$$\mathbf{S}(\mathbf{x}_{1:4}) = \begin{bmatrix} S_1(x_1) \\ S_2(x_1, x_2) \\ S_3(\cancel{x_1}, x_2, x_3) \\ S_4(\cancel{x_1}, \cancel{x_2}, x_3, x_4) \end{bmatrix} = \begin{bmatrix} S_1(x_1) \\ S_2(x_1, x_2) \\ S_3(x_2, x_3) \\ S_4(x_3, x_4) \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} = \mathbf{z}_{1:4}. \tag{15}$$

Equivalently, its inverse map would be:

$$\mathbf{S}^{-1}(\mathbf{z}_{1:4}) = \begin{bmatrix} S_1^{-1}(z_1) \\ S_2^{-1}(z_2; x_1) \\ S_3^{-1}(z_3; \cancel{x_1}, x_2) \\ S_4^{-1}(z_4; \cancel{x_1}, \cancel{x_2}, x_3) \end{bmatrix} = \begin{bmatrix} S_1^{-1}(z_1) \\ S_2^{-1}(z_2; x_1) \\ S_3^{-1}(z_3; x_2) \\ S_4^{-1}(z_4; x_3) \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \mathbf{x}_{1:4}. \tag{16}$$

Making use of conditional independence properties in this way is useful for two reasons:

1. **Robustness**: Any conditional independence we can enforce by construction is statistical information the map does not have to pry from the samples or the model, improving the overall fidelity and robustness of the approximation to π in settings with finite ensemble size.
2. **Efficiency**: The removal of superfluous dependencies reduces the number of input arguments to many of the map components S_k, decreasing the evaluation and inversion complexity (e.g., see Section 3.1.1 for evaluation complexity). This property is often called sparsification as it turns the Jacobian $\nabla\mathbf{S}$ into a sparse matrix.

This second property is the key to applying transport methods in high-dimensional systems. As each map component function S_k generically depends on all previous arguments $\mathbf{x}_{1:k-1}$, the computational demand explodes with the dimension of the target π when optimizing or evaluating the map. With sufficient conditional independence, however, sparse maps can overcome this dramatic increase in complexity. For instance, the Markov structure in Equations (15) and (16) results in a sparse map $\mathbf{S}$ with numerical complexity scaling linearly in the target dimension K.

A comprehensive account of the link between conditional independence and the sparsity of triangular maps is given in Spantini et al. (2018), through two main lines of results. First, given a sparse undirected probabilistic graphical model that encodes Markov properties of the target distribution π, it is shown how to predict the sparsity pattern of the triangular map $\mathbf{S}$. This process relies on an ordered graph elimination algorithm, and can thus be performed before learning the map itself. But the resulting sparsity pattern depends on the chosen ordering of the random variables, which underscores the fact that triangular maps are intrinsically *anisotropic* objects: a good ordering is necessary in order to maximize sparsity. We comment further on methods for finding such orderings in Section 5. Second, Spantini et al. (2018) show that a property somehow dual to sparsity is *decomposability*: given the Markov structure of some distribution π on $\mathbb{R}^K$, the inverse of the map $\mathbf{S}$ defined by $\mathbf{S}_\sharp\pi = \eta$ can be represented as the composition of finitely many low-dimensional triangular transport maps, where low dimensionality here means that each component map is a function only of a small number of inputs. The exact structure of this decomposition follows from, again, an ordered decomposition of the original graph. These results allow very high-dimensional problems to be broken into many smaller and more manageable parts, given some conditional independence.

2.3.4 Block-triangular maps

Recalling the block structure in Section 2.3.2, we have so far assumed that the map component "blocks" in Equation (11) are internally triangular; i.e., $\mathbf{S}_{1:k}$ has output $[S_1(x_1), \ldots, S_k(\mathbf{x}_{1:k})]$ and similar for $\mathbf{S}_{k+1:K}$. We may instead use *any* invertible functions $\mathbf{S}_{1:k} : \mathbb{R}^k \to \mathbb{R}^k$ and $\mathbf{S}_{k+1:K} : \mathbb{R}^K \to \mathbb{R}^{K-k}$, for example certain neural networks (e.g., Baptista et al., 2024c). This still permits sampling conditionals and can be numerically advantageous in high-dimensional