# Characteristic Circuit

**Zhongjie Yu**[1] **Martin Trapp**[2] **Kristian Kersting**[1,3,4]

[1]Department of Computer Science, TU Darmstadt, Germany
[2]Department of Computer Science, Aalto University, Finland
[3]Centre for Cognitive Science, TU Darmstadt, and Hessian Center for AI (hessian.AI)
[4]German Research Center for Artificial Intelligence (DFKI)

## Abstract

In many real-world scenarios it is crucial to be able to *reliably* and *efficiently* reason under uncertainty while capturing complex relationships in data. Probabilistic circuits (PCs), a prominent family of *tractable* probabilistic models, offer a remedy to this challenge by composing simple, tractable distributions into a high-dimensional probability distribution. However, learning PCs on heterogeneous data is challenging and densities of some parametric distributions are not available in closed form, limiting their potential use. We introduce characteristic circuits (CCs), a family of tractable probabilistic models providing a unified formalization of distributions over heterogeneous data in the spectral domain. The one-to-one relationship between characteristic functions and probability measures enables us to learn high-dimensional distributions on heterogeneous data domains and facilitates efficient probabilistic inference even when no closed-form density function is available. We show that the structure and parameters of CCs can be learned efficiently from the data and find that CCs outperform state-of-the-art density estimators for heterogeneous data domains on common benchmark data sets.

## 1 INTRODUCTION

Probabilistic circuits (PCs) have gained increasing attention in the machine learning community as a promising modelling family that renders many probabilistic inferences tractable with little compromise in their expressivity. Their beneficial properties have prompted many successful applications in density estimation [*e.g.*, Peharz et al., 2020, Di Mauro et al., 2021, Dang et al., 2022, Correia et al., 2023] and in areas where probabilistic reasoning is key, for example, neuro-symbolic reasoning [Ahmed et al., 2022], certified fairness [Selvam et al., 2023], or causality [Zečević et al., 2021]. Moreover, recent works have explored ways of specifying PCs for more complex modelling scenarios, such as time-series [*e.g.*, Trapp et al., 2020, Yu et al., 2021b,a] or tractable representation of graphs [Wang et al., 2022].

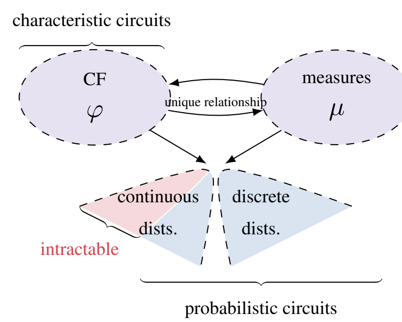


Figure 1: Characteristic circuits provide a unified, tractable specification of joint continuous and discrete distributions in the spectral domain.

Density estimation is at the very core of many machine learning techniques, *e.g.*, approximate Bayesian inference [Murphy, 2012], and a fundamental tool in statistics to identify characteristics of the data such as $n^{\text{th}}$ order moments or multimodality [Silverman, 2018]. However, even in the case of parametric families, densities are sometimes not available in closed-form, *e.g.*, only special cases of $\alpha$-stable distributions provide closed-form densities [Nolan, 2013]. Fortunately, there exists a one-to-one correspondence between probability measures and characteristic functions (CFs) [Sasvári, 2013], which can be understood as the Fourier-Stieltjes transform of the probability measures. Thus, enabling the characterisation of any probability measure through its CF. Henceforth, the CF of probability measures has found wide applicability in statistics, ranging from its use as a non-parametric estimator through the empirical characteristic function (ECF) [Feuerverger and Mureika, 1977] to estimate heavy-tailed data *e.g.*, through the family of $\alpha$-stable distributions [Nolan, 2013]. However, even though the CF

*Accepted for the 6th Workshop on Tractable Probabilistic Modeling at UAI* (TPM 2023).

has many advantageous properties and provides greater flexibility, its application to high-dimensional data and efficient computation of densities can be challenging [Nolan, 2013].

In this work, we bridge between the CF of probability measures and PCs, by examining PCs from a more general perspective, similar in spirit to their specifications as a summation over functions on a commutative semiring [Friesen and Domingos, 2016] or as a convex combination of product measures on product probability spaces [Trapp et al., 2020]. Most recently, Zhang et al. [2021] defined a circuit over probability generating polynomials to represent discrete probability distributions, resulting in a more expressive efficient model than existing families of tractable probabilistic models and achieving SOTA density estimation performance. Instead of defining the circuit over density functions, we propose to form the circuit over the *characteristic function* of the respective probability measures, illustrated in Fig. 1. Motivated by the property of characteristic function that the moments can be easily obtained by differentiating, we notice learning a circuit in the spectral domain has many interesting benefits: (i) *characteristic functions* as the base enables a *unified view* for discrete and continuous random variables (RVs), (ii) directly representing the *characteristic function* allows to learn distributions that *do not* have closed-form expressions for their density.

In summary, our contributions are: (1) We propose characteristic circuits, a novel deep probabilistic model representing the joint of discrete and continuous RVs through a unifying view in the spectral domain. (2) Moreover, we show that characteristic circuits retain tractability of PCs despite the change of domain and enable efficient computation of densities, marginals, and conditionals. (3) Lastly, we derive parameter and structure learning for characteristic circuits and find that characteristic circuits outperform SOTA density estimators in the majority of tested benchmarks.

## 2 PRELIMINARIES ON PROBABILISTIC CIRCUITS AND CHARACTERISTIC FUNCTIONS

Before introducing characteristic circuit, let us recap probabilistic circuits and characteristic functions.

**Probabilistic Circuits.** PCs are tractable probabilistic models, structured as rooted directed acyclic graphs, where each *leaf* node $\mathsf{L}$ represents a probability distribution over a univariate RV, each *sum* node $\mathsf{S}$ models a mixture of its children, and each *product* node $\mathsf{P}$ models a product distribution (assuming independence) of their children. A PC over a set of RVs $\boldsymbol{X}$ can be viewed as a computational graph $\mathcal{G}$ representing a tractable probability distribution over $\boldsymbol{X}$, and the value obtained at the root node is the probability computed by the circuit. We refer to Choi et al. [2020] for more details.

Each node in $\mathcal{G}$ is associated with a subset of $\boldsymbol{X}$ called the scope of a node $\mathsf{N}$ and is denoted as $\psi(\mathsf{N})$. The scope of an inner node is the union of the scope of its children. Sum nodes compute a weighted sum of their children $\mathsf{S} = \sum_{\mathsf{N} \in \mathrm{ch}(\mathsf{S})} w_{\mathsf{S},\mathsf{N}} \mathsf{N}$, and product nodes compute the product of their children $\mathsf{P} = \prod_{\mathsf{N} \in \mathrm{ch}(\mathsf{P})} \mathsf{N}$, where $\mathrm{ch}(\cdot)$ denotes the children of a node. The weights $w_{\mathsf{S},\mathsf{N}}$ are generally assumed to be non-negative and normalized (sum up to one) at each sum node. We also assume the PC to be smooth (complete) and decomposable [Darwiche, 2003], where smooth requires all children of a sum node having the same scope, and decomposable means all children of a product node having pairwise disjoint scopes.

**Characteristic Functions.** CFs provide a *unified view* for discrete and continuous RVs through the Fourier–Stieltjes transform of their probability measures. Let $\boldsymbol{X}$ be a random vector, the CF of $\boldsymbol{X}$ for $\boldsymbol{t} \in \mathbb{R}^d$ is given as:

$$\varphi_{\boldsymbol{X}}(\boldsymbol{t}) = \mathbb{E}[\exp(\mathrm{i}\, \boldsymbol{t}^\top \boldsymbol{X})] = \int_{\boldsymbol{x} \in \mathbb{R}^d} \exp(\mathrm{i}\, \boldsymbol{t}^\top \boldsymbol{x})\, \mu_{\boldsymbol{X}}(\mathrm{d}\boldsymbol{x}),$$

where $\mu_{\boldsymbol{X}}$ is the distribution/probability measure of $\boldsymbol{X}$. CFs have certain useful properties. We will briefly review those that are relevant for the remaining discussion: (i) $\varphi_X(0) = 1$ and $|\varphi_X(t)| \leq 1$; (ii) for any two RVs $X_1$, $X_2$, both have the same distribution iff $\varphi_{X_1} = \varphi_{X_2}$; (iii) if $X$ has $k$ moments, then $\varphi_X$ is $k$-times differentiable; and (iv) two RVs $X_1, X_2$ are independent iff $\varphi_{X_1,X_2}(s,t) = \varphi_{X_1}(s)\varphi_{X_2}(t)$. We refer to Sasvári [2013] for more details of CFs.

**Theorem 2.1** (Lévy's inversion theorem)**.** *Let $X$ be a real-valued random variable, $\mu_X$ its probability measure, and $\varphi_X : \mathbb{R} \to \mathbb{C}$ its characteristic function. Then for any $a, b \in \mathbb{R}$, $a < b$, we have that*

$$\lim_{T \to \infty} \frac{1}{2\pi} \int_{-T}^{T} \frac{\exp(-\mathrm{i}ta) - \exp(-\mathrm{i}tb)}{\mathrm{i}t} \varphi_X(t)\mathrm{d}t = \mu_X[(a,b)] + \frac{1}{2}(\mu_X(a) + \mu_X(b)) \tag{1}$$

*and, hence, $\varphi_X$ uniquely determines $\mu_X$.*

**Corollary.** *If $\int_{\mathbb{R}} |\varphi_X(t)|\mathrm{d}t < \infty$, then $X$ has a continuous probability density function $f_x$ given by*

$$f_X(x) = \frac{1}{2\pi} \int_{\mathbb{R}} \exp(-\mathrm{i}tx)\varphi_X(t)\mathrm{d}t. \tag{2}$$

Note that not every probability measure admits an analytical solution to Eq. (2), *e.g.*, only special cases of $\alpha$-stable distributions have a closed-form density function [Nolan, 2013], and numerical integration might be needed.

## 3 CHARACTERISTIC CIRCUIT

Now we have everything at hand to derive characteristic circuit. We first give a recursive definition of CC, followed by devising each type of node in a CC. We then show CCs feature efficient computation of densities, and in the end introduce how to learn a CC.

**Definition 3.1** (Characteristic Circuit). *Let $\boldsymbol{X} = \{X_1, \ldots, X_d\}$ be a set of random variables. A characteristic circuit denoted as $\mathcal{C}$ is a tuple consisting of a rooted directed acyclic graph $\mathcal{G}$, a scope function $\psi\colon \mathrm{V}(\mathcal{G}) \to \mathcal{P}(\boldsymbol{X})$, parameterized by a set of graph parameters $\theta_{\mathcal{G}}$. Nodes in $\mathcal{G}$ are either sum ($\mathsf{S}$), product ($\mathsf{P}$), or leaf ($\mathsf{L}$).*

*We define the characteristic circuit in the following recursive way: (1) a CF for a scalar random variable is a characteristic circuit, (2) a product of characteristic circuits is a characteristic circuit, (3) a convex combination of characteristic circuits is a characteristic circuit.*

Denote $\varphi_{\mathcal{C}}(\boldsymbol{t})$ the output of $\mathcal{C}$ computed at the root of $\mathcal{C}$ which represents the estimation of CF given argument of the CF $\boldsymbol{t} \in \mathbb{R}^d$. Further, we denote the number of RVs in the scope of $\mathsf{N}$ as $p_{\mathsf{N}} := |\psi(\mathsf{N})|$ and use $\varphi_{\mathsf{N}}(\boldsymbol{t})$ for the CF of a node. Throughout the paper we assume the CC to be smooth and decomposable.

**Product Nodes.** A product node in CC encodes the independence of its children. Let $X$ and $Y$ be two RVs. Following property (iv) of CFs, the CF of $X, Y$ is given as $\varphi_{X,Y}(t,s) = \varphi_X(t)\varphi_Y(s)$, if and only if $X$ and $Y$ are independent. Therefore, by definition, with $\boldsymbol{t} = \bigcup_{\mathsf{N}\in\mathrm{ch}(\mathsf{P})} \boldsymbol{t}_{\psi(\mathsf{N})}$, the characteristic function of product nodes is given as:

$$\varphi_{\mathsf{P}}(\boldsymbol{t}) = \prod\nolimits_{\mathsf{N}\in\mathrm{ch}(\mathsf{P})} \varphi_{\mathsf{N}}(\boldsymbol{t}_{\psi(\mathsf{N})}). \tag{3}$$

**Sum Nodes.** A sum node in CC encodes the mixture of its children. Let the parameters of $\mathsf{S}$ be given as $\sum_{\mathsf{N}\in\mathrm{ch}(\mathsf{S})} w_{\mathsf{S},\mathsf{N}} = 1$ and $w_{\mathsf{S},\mathsf{N}} \geq 0, \forall \mathsf{S}, \mathsf{N}$. Then the sum node in a CC is given as: $\varphi_{\mathsf{S}}(\boldsymbol{t}) =$

$$\int_{\boldsymbol{x}\in\mathbb{R}^d} \exp(\mathrm{i}\boldsymbol{t}^\top\boldsymbol{x}) \left[\sum\nolimits_{\mathsf{N}\in\mathrm{ch}(\mathsf{S})} w_{\mathsf{S},\mathsf{N}}\, \mu_{\mathsf{N}}(\mathrm{d}\boldsymbol{x})\right]$$
$$= \sum\nolimits_{\mathsf{N}\in\mathrm{ch}(\mathsf{S})} w_{\mathsf{S},\mathsf{N}} \underbrace{\int_{\boldsymbol{x}\in\mathbb{R}^{p_{\mathsf{S}}}} \exp(\mathrm{i}\boldsymbol{t}^\top\boldsymbol{x})\, \mu_{\mathsf{N}}(\mathrm{d}\boldsymbol{x})}_{=\varphi_{\mathsf{N}}(\boldsymbol{t})}. \tag{4}$$

**Leaf Nodes.** The leaf node of a CC models the characteristic function of a univariate RV. For discrete RVs, we utilize categorical distributions and for continuous RVs, we use either normal distribution or $\alpha$-stable distributions. Besides, ECF leaf $\varphi_{\mathsf{L}_{\mathrm{ECF}}}(t) = \frac{1}{n}\sum_{j=1}^{n} \exp(\mathrm{i}\, t\, x_j)$ is also employed as a non-parametric leaf. A more detailed discussion on leaf types can be found in Appendix A.

### 3.1 THEORETIC PROPERTIES

With everything at hand, we can derive the marginal density.

**Efficient Computation of Densities.** Through their recursive nature, CCs enable efficient computation of densities in high-dimensional settings even if the density function is not available in closed form. For this, we present an extension of Theorem 2.1 for CCs, formulated using the notion of induced trees $\mathcal{T}$ [Zhao et al., 2016].

**Lemma 3.2** (Inversion). *Let $\mathcal{C} = \langle \mathcal{G}, \psi, \theta_{\mathcal{G}} \rangle$ be a CC on RVs $\boldsymbol{X} = \{X_j\}_{j=1}^d$ with univariate leave nodes. If $\int_{\mathbb{R}} |\varphi_{\mathsf{L}}(t)| \mathrm{d}t < \infty$ for every $\mathsf{L} \in V(\mathcal{G})$, then $\boldsymbol{X}$ has a continuous probability density function $f_{\boldsymbol{x}}$ given by $f_{\boldsymbol{X}}(\boldsymbol{x}) =$*

$$\frac{1}{(2\pi)^d} \sum_{i=1}^{\tau} \prod_{(\mathsf{S},\mathsf{N})\in\mathrm{E}(\mathcal{T}_i)} w_{\mathsf{S},\mathsf{N}} \prod_{\mathsf{L}\in\mathrm{V}(\mathcal{T}_i)} \int_{\mathbb{R}} \exp(-\mathrm{i}t x_{\psi(\mathsf{L})}) \varphi_{\mathsf{L}}(t)\, \mathrm{d}t,$$

*and can be computed efficiently through analytic or numerical integration at the leaves. Proof is in Appendix B.*

**Computation of Marginals.** Similar to PCs over distribution functions, CCs allow efficient computation of arbitrary marginals. Given a CC on RVs $\boldsymbol{Z} = \boldsymbol{X} \cup \boldsymbol{Y}$, we can obtain the marginal CC over $\boldsymbol{X}$ as follows. Let $n = |\boldsymbol{X}|$, $m = |\boldsymbol{Y}|$, $\boldsymbol{t} = \boldsymbol{t}_{\boldsymbol{X}} \cup \boldsymbol{t}_{\boldsymbol{Y}} \in \mathbb{R}^{n+m}$ and let the characteristic function of the circuit be given by $\varphi_{\mathcal{C}}(\boldsymbol{t}_{\boldsymbol{X}}, \boldsymbol{t}_{\boldsymbol{Y}}) = \int_{\boldsymbol{x}\in\mathbb{R}^{n+m}} \exp(\mathrm{i}\boldsymbol{t}^\top\boldsymbol{x})\mu_{\mathsf{S}}(\mathrm{d}\boldsymbol{x})$, where $\mu_{\mathsf{S}}$ denotes the distribution of the root. Then following property (i) of CFs, the marginal CC over $\boldsymbol{X}$ is given by setting $\boldsymbol{t}_{\boldsymbol{Y}} = \boldsymbol{0}$.

**Algorithm 1** CC Structure Learning

```
Input: training data D, RVs X, threshold min_k, number of splits k_S and k_P
Output: C
Function buildCF(D, X)
    return L ← univariate CF φ_X(t)
Function buildSumNode(D, X)
    if |X| = 1 then
        S ← buildCF(D, X) ;
    else if |D| ≤ min_k then
        Partition D into |X| independent subsets D_j ;
        S ← ∏_{j=1}^{|X|} buildCF(D_j, X_j) ;
    else
        Partition D into k_S clusters D_i ;
        S ← ∑_{i=1}^{k_S} (|D_i|/|D|) buildProdNode(D_i, X) ;
    return S
Function buildProdNode(D, X)
    Partition D into k_P independent subsets D_j ;
    return P ← ∏_{j=1}^{k_P} buildSumNode(D_j, X_j)
C ← buildSumNode(D, X)
```

### 3.2 LEARNING CHARACTERISTIC CIRCUIT

Inspired by Gens and Pedro [2013], we propose a *structure learning* algorithm to learn the structure of the CC, depicted in Algorithm 1. Structure learning recursively splits the data slice and creates sum and product nodes of the CC. Univariate CF leaves are created to model the local data.

On the other hand, given a random circuit structure, *parameter learning* can be employed to learn a CC. Instead of maximising the likelihood, which is not guaranteed to be tractable, we minimise the approximated squared characteristic function distance (CFD, c.f. Appendix C) between the

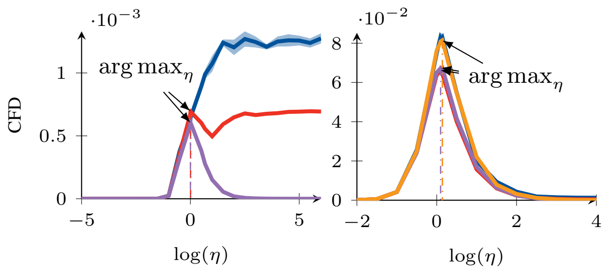

Figure 2: CC approximates the true distribution better than the ECF by providing a smaller CFD. We visualize the CFD for CC with CC-P ( ), CC-E ( ), CC-N ( ) and a single ECF ( ) learned from MM (Left) and BN (Right).

CC and ECF: $\frac{1}{k}\sum_{j=1}^{k}|\frac{1}{n}\sum_{i=1}^{n}\exp(\mathrm{i}\boldsymbol{t}_j^\top\boldsymbol{x}_i)-\varphi_{\mathcal{C}}(\boldsymbol{t}_j)|^2$, where $\boldsymbol{t}_j \sim \omega(\boldsymbol{t};\eta)$, in this paper $\mathcal{N}(\boldsymbol{0},\mathrm{diag}(\eta^2))$. Furthermore, we provide an analytical solution to the CFD between two CCs in Appendix D.

# 4 EXPERIMENTAL EVALUATION

Here, we evaluate the performance of characteristic circuit on synthetic data sets and the UCI data sets, which are composed of heterogeneous data.

**Better Approximation of Characteristic Functions**. We begin by describing and evaluating the performance of CC on two synthetic data sets: a mixture of multivariate distributions (denoted as MM): $p(\boldsymbol{x}) = \sum_{i=1}^{K} w_i p(\boldsymbol{x}_1 \mid \mu_i, \sigma_i^2) p(\boldsymbol{x}_2 \mid \boldsymbol{p}_i)$, where $p(\boldsymbol{x} \mid \mu, \sigma^2)$ is the univariate normal distribution and $p(\boldsymbol{x} \mid \boldsymbol{p})$ is the univariate categorical distribution, and a Bayesian network with 5 nodes (1 continuous and 4 discrete nodes, denoted as BN).

We first learn CCs applying structure learning with various leaf types: CC with ECF as leaves (CC-E), CC with normal distribution for continuous RVs and categorical distributions for discrete RVs, *i.e.*, parametric leaves (CC-P), and CC with normal distribution for all leaf nodes (CC-N). The trained CCs are evaluated with the CFD between the CC and the ground truth CF. Following Chwialkowski et al. [2015] and Ansari et al. [2020], we illustrate both the CFD with varying scale $\eta$ in $\omega(\boldsymbol{t};\eta)$ and also optimising $\eta$ for the largest CFD, shown in Fig. 2. The CFDs are averaged for 5 runs and the standard deviations are visualized. Fig. 2 shows that both CC-E and CC-P have almost equally lower CFD values and also lower maximum CFD values compared to the ECF, which indicates the CC structure better encodes the data distribution than the ECF. Therefore, CC estimates the data distribution better than ECF, which can be justified by the smaller CFD from CC-E compared with ECF.

**Better Density Estimator on Heterogeneous Data.** Real-world tabular data usually contain both discrete and real-valued elements, and thus are in most cases heterogeneous data. Therefore, we also conduct experiments on the UCI heterogeneous data sets [Molina et al., 2018] and compare

Table 1: Average test log-likelihoods from CC and SOTA algorithms on heterogeneous data. CC with $\alpha$-stable distribution leaves (CC-A) wins 7 out of 12 data sets.

| Data Set | MSPN | ABDA | BSPN | CC-P | CC-A |
|---|---|---|---|---|---|
| Abalone | **9.73** | 2.22 | 3.92 | 4.27 | 3.24 |
| Adult | -44.07 | -5.91 | **-4.62** | -31.37 | -8.65 |
| Australian | -36.14 | -16.44 | -21.51 | -30.29 | **-3.25** |
| Autism | -39.20 | -27.93 | **-0.47** | -34.70 | -13.56 |
| Breast | -28.01 | -25.48 | -25.02 | -54.75 | **-13.22** |
| Chess | -13.01 | -12.30 | **-11.54** | -13.03 | -13.03 |
| Crx | -36.26 | -12.82 | -19.38 | -32.62 | **-3.80** |
| Dermatology | -27.71 | -24.98 | **-23.95** | -30.33 | -24.72 |
| Diabetes | -31.22 | -17.48 | -21.21 | -23.00 | **-0.73** |
| German | -26.05 | -25.83 | -26.76 | -27.29 | **-14.96** |
| Student | -30.18 | -28.73 | -29.51 | -31.59 | **-27.79** |
| Wine | -0.13 | -10.12 | -8.62 | -6.91 | **3.49** |
| # best | 1 | 0 | 4 | 0 | 7 |

against SOTA PC methods, including MSPN [Molina et al., 2018], ABDA [Vergari et al., 2019] and BSPN [Trapp et al., 2019], for density estimation. We employ structure learning and all the continuous RVs are modelled with either normal distribution (CC-P) or $\alpha$-stable distribution (CC-A).

The test log-likelihoods are presented in Table 1. CC-P does not win on all the data sets but performs as runner-up on Abalone, and is also competitive with MSPN and ABDA on most of the data sets. CC-A outperforms the other methods on 7 out of 12 data sets, marked as bold. This implies that CC is a competitive density estimator compared with SOTA PCs. Furthermore, $\alpha$-stable distribution leaf, which is not available in current PCs, is a more suitable choice for CC on heterogeneous tabular data.

# 5 CONCLUSION

We introduced characteristic circuit, the first circuit based characteristic function estimator that leverages an arithmetic circuit with univariate characteristic function leaves for modelling the joint of heterogeneous data distributions. Compared to PCs, CC models the CF of data distribution in the continuous spectral domain, providing a unified view for discrete and continuous RVs, and can further model distributions that do not have closed-form probability density functions. We show that both joint and marginal probability densities can be calculated exactly and efficiently via CC. Finally, we empirically show that CC approximates data distribution better than ECF, and can also perform as a competitive density estimator on heterogeneous data sets.

The circuit structure of CC generated by structure learning has a high impact on the performance of the CC, thus an inappropriate structure can limit the modelling power of CC. Therefore, one interesting direction of future work is to apply parameter learning of CC on more advanced circuit structures [Peharz et al., 2020] or in combinations with flows.

**Acknowledgements**

This work was supported by the Federal Ministry of Education and Research (BMBF) Competence Center for AI and Labour ("kompAKI", FKZ 02L19C150). It benefited from the Hessian Ministry of Higher Education, Research, Science and the Arts (HMWK; projects "The Third Wave of AI" and "The Adaptive Mind"), and the Hessian research priority programme LOEWE within the project "WhiteBox". MT acknowledges funding from the Academy of Finland (grant number 347279).

## References

Kareem Ahmed, Stefano Teso, Kai-Wei Chang, Guy Van den Broeck, and Antonio Vergari. Semantic probabilistic layers for neuro-symbolic learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Abdul Fatir Ansari, Jonathan Scarlett, and Harold Soh. A characteristic function approach to deep implicit generative modeling. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7478–7487, 2020.

Y Choi, Antonio Vergari, and Guy Van den Broeck. Probabilistic circuits: A unifying framework for tractable probabilistic models. Technical report, UCLA, 2020.

Kacper P Chwialkowski, Aaditya Ramdas, Dino Sejdinovic, and Arthur Gretton. Fast two-sample testing with analytic representations of probability measures. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 28, 2015.

Alvaro HC Correia, Gennaro Gala, Erik Quaeghebeur, Cassio de Campos, and Robert Peharz. Continuous mixtures of tractable probabilistic models. In *AAAI Conference on Artificial Intelligence*, 2023.

Harald Cramér. *Mathematical methods of statistics*, volume 26. Princeton university press, 1999.

Meihua Dang, Antonio Vergari, and Guy Van den Broeck. Strudel: A fast and accurate learner of structured-decomposable probabilistic circuits. *International Journal of Approximate Reasoning*, 140:92–115, 2022.

Adnan Darwiche. A differential approach to inference in bayesian networks. *Journal of the ACM (JACM)*, 50(3): 280–305, 2003.

Nicola Di Mauro, Gennaro Gala, Marco Iannotta, and Teresa MA Basile. Random probabilistic circuits. In *Uncertainty in Artificial Intelligence (UAI)*, pages 1682–1691, 2021.

Andrey Feuerverger and Roman A Mureika. The empirical characteristic function and its applications. *The annals of Statistics*, pages 88–97, 1977.

Abram Friesen and Pedro Domingos. The sum-product theorem: A foundation for learning tractable models. In *International Conference on Machine Learning (ICML)*, pages 1909–1918, 2016.

Robert Gens and Domingos Pedro. Learning the structure of sum-product networks. In *International Conference on Machine Learning (ICML)*, 2013.

Alejandro Molina, Antonio Vergari, Nicola Di Mauro, Sriraam Natarajan, Floriana Esposito, and Kristian Kersting. Mixed sum-product networks: A deep architecture for hybrid domains. In *AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

John P Nolan. Multivariate elliptically contoured stable distributions: theory and estimation. *Computational statistics*, 28:2067–2089, 2013.

Robert Peharz, Steven Lang, Antonio Vergari, Karl Stelzner, Alejandro Molina, Martin Trapp, Guy Van den Broeck, Kristian Kersting, and Zoubin Ghahramani. Einsum networks: Fast and scalable learning of tractable probabilistic circuits. In *International Conference on Machine Learning (ICML)*, pages 7563–7574, 2020.

Zoltán Sasvári. *Multivariate characteristic and correlation functions*, volume 50. Walter de Gruyter, 2013.

Nikil Roashan Selvam, Guy Van den Broeck, and YooJung Choi. Certifying fairness of probabilistic circuits. In *AAAI Conference on Artificial Intelligence*, 2023.

Bernard W Silverman. *Density estimation for statistics and data analysis*. Routledge, 2018.

Bharath K Sriperumbudur, Arthur Gretton, Kenji Fukumizu, Bernhard Schölkopf, and Gert RG Lanckriet. Hilbert space embeddings and metrics on probability measures. *The Journal of Machine Learning Research*, 11:1517–1561, 2010.

Martin Trapp, Robert Peharz, Hong Ge, Franz Pernkopf, and Zoubin Ghahramani. Bayesian learning of sum-product networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019.

Martin Trapp, Robert Peharz, Franz Pernkopf, and Carl Edward Rasmussen. Deep structured mixtures of gaussian processes. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 2251–2261, 2020.

Antonio Vergari, Alejandro Molina, Robert Peharz, Zoubin Ghahramani, Kristian Kersting, and Isabel Valera. Automatic bayesian density analysis. In *AAAI Conference on Artificial Intelligence*, volume 33, pages 5207–5215, 2019.

Benjie Wang, Matthew R Wicker, and Marta Kwiatkowska. Tractable uncertainty for structure learning. In *International Conference on Machine Learning (ICML)*, pages 23131–23150, 2022.

Zhongjie Yu, Fabrizio Ventola, and Kristian Kersting. Whittle networks: A deep likelihood model for time series. In *International Conference on Machine Learning (ICML)*, pages 12177–12186, 2021a.

Zhongjie Yu, Mingye Zhu, Martin Trapp, Arseny Skryagin, and Kristian Kersting. Leveraging probabilistic circuits for nonparametric multi-output regression. In *Uncertainty in Artificial Intelligence (UAI)*, 2021b.

Matej Zečević, Devendra Singh Dhami, Athresh Karanam, Sriraam Natarajan, and Kristian Kersting. Interventional sum-product networks: Causal inference with tractable probabilistic models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Honghua Zhang, Brendan Juba, and Guy Van den Broeck. Probabilistic generating circuits. In *International Conference on Machine Learning (ICML)*, pages 12447–12457, 2021.

Han Zhao, Tameem Adel, Geoff Gordon, and Brandon Amos. Collapsed variational inference for sum-product networks. In *International Conference on Machine Learning (ICML)*, pages 1310–1318, 2016.

# Supplementary Material:
# Characteristic Circuit

## A LEAF TYPES OF CHARACTERISTIC CIRCUIT

Here we describe the leaf types that are used in the characteristic circuit.

*ECF leaf.* In many cases, a parametric form of the data distribution is not available and one needs to use a non-parametric estimator. The ECF [Feuerverger and Mureika, 1977, Cramér, 1999] is an unbiased and consistent non-parametric estimator of the population characteristic function. Given data $\{\boldsymbol{x}_j\}_{j=1}^n$ the ECF is given by $\hat{\varphi}_{\mathbb{P}}(\boldsymbol{t}) = \frac{1}{n}\sum_{j=1}^n \exp(\mathrm{i}\,\boldsymbol{t}^\top \boldsymbol{x}_j)$. Thus, the most straightforward way for modelling the leaf node is to directly employ the empirical characteristic function for the local data at each leaf, defined as $\varphi_{\mathsf{L}_{\text{ECF}}}(t) = \frac{1}{n}\sum_{j=1}^n \exp(\mathrm{i}\,t\,x_j)$, where $n$ is the number of instances at leaf $\mathsf{L}$, and $x_j$ is the $j^{th}$ instance. The ECF leaf is non-parametric and is determined by the $n$ instances $x_j$ at $\mathsf{L}$.

*Parametric leaf for continuous RVs.* Motivated by existing SPN literature, we can assume that the RV at a leaf node follows a parametric continuous distribution *e.g.* normal distribution. With this, the leaf node is equipped with the CF of normal distribution $\varphi_{\mathsf{L}_{\text{Normal}}}(t) = \exp(\mathrm{i}\,t\,\mu - \frac{1}{2}\sigma^2 t^2)$, where parameters $\mu$ and $\sigma^2$ are the mean and variance.

*Parametric leaf for discrete RVs.* For discrete RVs, if it is assumed to follow categorical distribution ($P(X=j)=p_j$), then the CF at the leaf node can be defined as $\varphi_{\mathsf{L}_{\text{Categorical}}}(t) = \mathbb{E}[\exp(\mathrm{i}\,t\,x)] = \sum_{j=1}^k p_j \exp(\mathrm{i}\,t\,j)$. Other discrete distributions which are widely used in probabilistic circuits can also be employed as leaf nodes in CCs, *e.g.* Bernoulli, Poisson and geometric distributions.

*$\alpha$-stable leaf.* In the case of financial data or data distributed with heavy tails, the $\alpha$-stable distribution is frequently employed. $\alpha$-stable distributions are more flexible in modelling *e.g.* data with skewed centered distributions. The characteristic function of an $\alpha$-stable distribution is $\varphi_{\mathsf{L}_{\alpha\text{-stable}}}(t) = \exp(\mathrm{i}\,t\,\mu - |ct|^\alpha (1 - \mathrm{i}\beta \mathrm{sgn}(t)\Phi))$, where $\mathrm{sgn}(t)$ takes the sign of $t$ and $\Phi = \begin{cases} \tan(\pi\alpha/2) & \alpha \neq 1 \\ -2/\pi \log|t| & \alpha = 1 \end{cases}$. The parameters in $\alpha$-stable distribution are the stability parameter $\alpha$, the skewness parameter $\beta$, the scale parameter $c$ and the location parameter $\mu$. Despite its modelling power, $\alpha$-stable distribution is never employed in PCs, as it is represented analytically by its CF and in most cases does not have a closed-form probability density function.

## B THEORETIC PROPERTIES

In this section, we provide a detailed proof of Lemma 3.2

*Proof.* Let $\mathcal{C} = \langle \mathcal{G}, \psi, \theta_{\mathcal{G}} \rangle$ be a characteristic circuit on RVs $\boldsymbol{X} = \{X_j\}_{j=1}^d$ with univariate leave nodes and $p_{\mathsf{N}}$ the number of RVs in the scope of $\mathsf{N}$. In order to calculate the density function of $\mathcal{C}$, we need to integrate over the $d$-dimensional real space $\mathbb{R}^d$, *i.e.*,

$$f_{\boldsymbol{X}}(\boldsymbol{x}) = \frac{1}{(2\pi)^d} \underbrace{\int_{\boldsymbol{t}\in\mathbb{R}^d} \exp(-\mathrm{i}\,\boldsymbol{t}^\top \boldsymbol{x})\, \varphi_{\mathcal{C}}(\boldsymbol{t})\, \lambda_d(\mathrm{d}\boldsymbol{t})}_{=\hat{f}_{\mathcal{C}}(\boldsymbol{x})}, \tag{5}$$

where $\varphi_{\mathcal{C}}(\boldsymbol{t})$ denotes the CF defined by the root of the characteristic circuit and $\lambda_d$ is the Lebesque measure on $(\mathbb{R}^d, \mathcal{B}(\mathbb{R}^d))$. We can examine the computation of Eq. (5) recursively for every node.

**Leaf Nodes.** If $\mathsf{N}$ is a leaf node $\mathsf{L}$, we obtain $\hat{f}_{\mathsf{N}}(\cdot)$ by calculating:

$$\hat{f}_{\mathsf{L}}(x) = 2\pi f_{\mathsf{L}}(x) = \int_{\mathbb{R}} \exp(-\mathrm{i}tx)\varphi_X(t)\lambda(\mathrm{d}t), \tag{6}$$

which follows from Theorem 2.1.

**Sum Nodes.** If $\mathsf{N}$ is a sum node $\mathsf{S}$, then:

$$\hat{f}_{\mathsf{S}}(\boldsymbol{x}) = \int_{\boldsymbol{t}\in\mathbb{R}^p} \exp(-\mathrm{i}\,\boldsymbol{t}^\top \boldsymbol{x})\,\varphi_{\mathsf{S}}(\boldsymbol{t})\,\lambda_p(\mathrm{d}\boldsymbol{t}) = \sum_{\mathsf{N}\in\mathrm{ch}(\mathsf{S})} w_{\mathsf{S},\mathsf{N}} \underbrace{\int_{\boldsymbol{t}\in\mathbb{R}^{p_{\mathsf{S}}}} \exp(-\mathrm{i}\,\boldsymbol{t}^\top \boldsymbol{x})\,\varphi_{\mathsf{N}}(\boldsymbol{t})\,\lambda_{p_{\mathsf{S}}}(\mathrm{d}\boldsymbol{t})}_{=\hat{f}_{\mathsf{N}}(\boldsymbol{x})}. \tag{7}$$

Therefore, computing the inverse for $\mathsf{S}$ reduces to inversion at its children.

**Product Nodes.** If $\mathsf{N}$ is a product node $\mathsf{P}$, then:

$$\hat{f}_{\mathsf{P}}(\boldsymbol{x}) = \int_{\boldsymbol{t}\in\mathbb{R}^{p_{\mathsf{P}}}} \exp(-\mathrm{i}\,\boldsymbol{t}^\top \boldsymbol{x})\,\varphi_{\mathsf{P}}(\boldsymbol{t})\,\lambda_{p_{\mathsf{P}}}(\mathrm{d}\boldsymbol{t}) = \prod_{\mathsf{N}\in\mathrm{ch}(\mathsf{P})} \underbrace{\int_{\boldsymbol{s}\in\mathbb{R}^{p_{\mathsf{N}}}} \exp(-\mathrm{i}\,\boldsymbol{s}^\top \boldsymbol{x}_{[\psi(\mathsf{N})]})\,\varphi_{\mathsf{N}}(\boldsymbol{s})\,\lambda_{p_{\mathsf{N}}}(\mathrm{d}\boldsymbol{s})}_{=\hat{f}_{\mathsf{N}}(\boldsymbol{x}_{[\psi(\mathsf{N})]})}, \tag{8}$$

where we used that $\lambda_{p_{\mathsf{P}}} = \otimes_{\mathsf{N}\in\mathrm{ch}(\mathsf{P})}\lambda_{p_{\mathsf{N}}}$ is a product measure on a product space, applied Fubini's theorem, and used the additivity property of exponential functions. Consequently, computing the inverse for $\mathsf{P}$ reduces to inversion at its children.

Through the recursive application of Eq. (7) and Eq. (8), we obtain that Eq. (5) reduces to integration at the leaves and, therefore, can be solved either analytically or efficiently through one-dimensional numerical integration. $\square$

## C CHARACTERISTIC FUNCTION DISTANCE

To measure the distance between two distributions represented by their characteristic functions, the squared characteristic function distance (CFD) can be employed. The CFD between two distributions $\mathbb{P}$ and $\mathbb{Q}$ is defined as:

$$\mathrm{CFD}^2_\omega(\mathbb{P},\mathbb{Q}) = \int_{\mathbb{R}^d} |\varphi_{\mathbb{P}}(\boldsymbol{t}) - \varphi_{\mathbb{Q}}(\boldsymbol{t})|^2\,\omega(\boldsymbol{t};\eta)\mathrm{d}\boldsymbol{t}, \tag{9}$$

where $\omega(\boldsymbol{t};\eta)$ is a weighting function parameterized by $\eta$ and guarantees the integral in Eq. (9) converge. When $\omega(\boldsymbol{t};\eta)$ is a probability density function, Eq. (9) can be rewritten as:

$$\mathrm{CFD}^2_\omega(\mathbb{P},\mathbb{Q}) = \mathbb{E}_{\boldsymbol{t}\sim\omega(\boldsymbol{t};\eta)}\left[|\varphi_{\mathbb{P}}(\boldsymbol{t}) - \varphi_{\mathbb{Q}}(\boldsymbol{t})|^2\right]. \tag{10}$$

Sriperumbudur et al. [2010] showed that using the uniqueness theorem of CFs, $\mathrm{CFD}_\omega(\mathbb{P},\mathbb{Q}) = 0$ iff $\mathbb{P}=\mathbb{Q}$. Computing Eq. (10) is generally intractable, therefore, we use Monte-Carlo integration to approximate the expectation, resulting in $\mathrm{CFD}^2_\omega(\mathbb{P},\mathbb{Q}) \approx \frac{1}{k}\sum_{j=1}^k |\varphi_{\mathbb{P}}(t_j) - \varphi_{\mathbb{Q}}(t_j)|^2$, where $\{t_1,\cdots,t_k\} \overset{\text{i.i.d.}}{\sim} \omega(\boldsymbol{t};\eta)$. We refer to Ansari et al. [2020] for a detailed discussion.

## D ANALYTICAL SOLUTION OF THE CFD

The squared characteristic function distance (CFD)

$$\mathrm{CFD}^2_\omega(\mathbb{P},\mathbb{Q}) = \int_{\mathbb{R}^d} |\varphi_{\mathbb{P}}(\boldsymbol{t}) - \varphi_{\mathbb{Q}}(\boldsymbol{t})|^2\,\omega(\boldsymbol{t};\eta)\mathrm{d}\boldsymbol{t} \tag{11}$$

can not only be estimated with MC methods by sampling from $\omega(\boldsymbol{t};\eta)$, but also be calculated through the characteristic circuit with an analytical solution, if $\varphi_{\mathbb{P}}(\boldsymbol{t})$ and $\varphi_{\mathbb{Q}}(\boldsymbol{t})$ are compatible characteristic circuits.

Eq. (11) can be rewritten as

$$\begin{aligned}\mathrm{CFD}^2_\omega(\mathbb{P},\mathbb{Q}) &= \int_{\mathbb{R}^d} (\varphi_{\mathbb{P}}(\boldsymbol{t}) - \varphi_{\mathbb{Q}}(\boldsymbol{t}))\left(\overline{\varphi_{\mathbb{P}}(\boldsymbol{t}) - \varphi_{\mathbb{Q}}(\boldsymbol{t})}\right)\omega(\boldsymbol{t};\eta)\mathrm{d}\boldsymbol{t} \\ &= \int_{\mathbb{R}^d} (\varphi_{\mathbb{P}}(\boldsymbol{t})\overline{\varphi_{\mathbb{P}}(\boldsymbol{t})} - \varphi_{\mathbb{P}}(\boldsymbol{t})\overline{\varphi_{\mathbb{Q}}(\boldsymbol{t})} - \varphi_{\mathbb{Q}}(\boldsymbol{t})\overline{\varphi_{\mathbb{P}}(\boldsymbol{t})} + \varphi_{\mathbb{Q}}(\boldsymbol{t})\overline{\varphi_{\mathbb{Q}}(\boldsymbol{t})})\omega(\boldsymbol{t};\eta)\mathrm{d}\boldsymbol{t},\end{aligned} \tag{12}$$

where $\overline{z}$ denotes the conjugate of the complex number $z$. Without loss of generality, let us derive the analytical solution of $\int_{\mathbb{R}^d} \varphi_{\mathbb{P}}(\boldsymbol{t})\overline{\varphi_{\mathbb{Q}}(\boldsymbol{t})}\omega(\boldsymbol{t};\eta)\mathrm{d}\boldsymbol{t}$, since the derivation can be directly applied to the other terms in Eq. (12). In the following, we omit the term $\omega(\boldsymbol{t};\eta)$ at sum and product nodes for simplicity. At sum nodes $\mathsf{S}$ and $\mathsf{S}'$,

$$\begin{aligned}\int \mathsf{S}(\boldsymbol{t})\overline{\mathsf{S}'(\boldsymbol{t})}\mathrm{d}\boldsymbol{t} &= \int \left(\sum_{\mathsf{N}\in\mathrm{ch}(\mathsf{S})} w_{\mathsf{S},\mathsf{N}}\mathsf{N}(\boldsymbol{t})\right)\left(\sum_{\mathsf{N}'\in\mathrm{ch}(\mathsf{S}')} w_{\mathsf{S}',\mathsf{N}'}\overline{\mathsf{N}'(\boldsymbol{t})}\right)\mathrm{d}\boldsymbol{t} \\ &= \int \sum_{\mathsf{N}\in\mathrm{ch}(\mathsf{S})}\sum_{\mathsf{N}'\in\mathrm{ch}(\mathsf{S}')} w_{\mathsf{S},\mathsf{N}}w_{\mathsf{S}',\mathsf{N}'}\mathsf{N}(\boldsymbol{t})\overline{\mathsf{N}'(\boldsymbol{t})}\mathrm{d}\boldsymbol{t} \\ &= \sum_{\mathsf{N}\in\mathrm{ch}(\mathsf{S})}\sum_{\mathsf{N}'\in\mathrm{ch}(\mathsf{S}')} w_{\mathsf{S},\mathsf{N}}w_{\mathsf{S}',\mathsf{N}'}\int \mathsf{N}(\boldsymbol{t})\overline{\mathsf{N}'(\boldsymbol{t})}\mathrm{d}\boldsymbol{t}.\end{aligned} \tag{13}$$

At product nodes $\mathsf{P}$ and $\mathsf{P}'$,

$$\begin{aligned}\int \mathsf{P}(\boldsymbol{t})\overline{\mathsf{P}'(\boldsymbol{t})}\mathrm{d}\boldsymbol{t} &= \int \left(\prod_{\mathsf{N}\in\mathrm{ch}(\mathsf{P})} \mathsf{N}(\boldsymbol{t}_{[\psi(\mathsf{N})]})\right)\left(\prod_{\mathsf{N}'\in\mathrm{ch}(\mathsf{P}')} \overline{\mathsf{N}'(\boldsymbol{t}_{[\psi(\mathsf{N}')]})}\right)\mathrm{d}\boldsymbol{t} \\ &\overset{\text{compatibility}}{=} \int \prod_{(\mathsf{N},\mathsf{N}')\in\Delta_{\mathsf{P}\times\mathsf{P}'}} \mathsf{N}(\underbrace{\boldsymbol{t}_{[\psi(\mathsf{N})]}}_{=\hat{\boldsymbol{t}}})\overline{\mathsf{N}'(\boldsymbol{t}_{[\psi(\mathsf{N}')]})}\,\mathrm{d}\boldsymbol{t}\end{aligned} \tag{14}$$

where $\Delta_{\mathsf{P}\times\mathsf{P}'}$ denotes the diagonal of the Cartesian product of the children of $\mathsf{P}$ and $\mathsf{P}'$, i.e., $\mathrm{diag}(\mathrm{ch}(\mathsf{P}) \times \mathrm{ch}(\mathsf{P}'))$, compatibility ensures that both product nodes apply the same partition of the scope $\psi(\mathsf{P}) = \psi(\mathsf{P}')$ with parts in the same order, and $\boldsymbol{t}_{[\psi(\mathsf{N})]}$ is the projection of $\boldsymbol{t}$ to the scope of $\mathsf{N}$. Therefore,

$$= \prod_{(\mathsf{N},\mathsf{N}')\in\Delta_{\mathsf{P}\times\mathsf{P}'}} \int_{\mathbb{R}^{p_\mathsf{N}}} \mathsf{N}(\hat{\boldsymbol{t}})\overline{\mathsf{N}'(\hat{\boldsymbol{t}})}\,\mathrm{d}\hat{\boldsymbol{t}}. \qquad \text{(compatibility)} \tag{15}$$

At univariate leaf nodes $\mathsf{L}$ and $\mathsf{L}'$, assuming both leaf nodes model univariate normal distribution with parameters $(\mu,\sigma)$ and $(\mu',\sigma')$, and $\omega(t;\eta) = \frac{1}{\eta\sqrt{2\pi}}\exp(\frac{-t^2}{2\eta^2})$, then

$$\begin{aligned}\int_{\mathbb{R}} \mathsf{L}(t)\overline{\mathsf{L}'(t)}\omega(t;\eta)\mathrm{d}t &= \int_{\mathbb{R}} \exp(\mathrm{i}\,t\,\mu - \frac{1}{2}\sigma^2 t^2)\overline{\exp(\mathrm{i}\,t\,\mu' - \frac{1}{2}\sigma'^2 t^2)}\frac{1}{\eta\sqrt{2\pi}}\exp(\frac{-t^2}{2\eta^2})\mathrm{d}t \\ &\overset{\overline{\mathrm{e}^{z}}=\mathrm{e}^{\overline{z}}}{=} \frac{1}{\eta\sqrt{2\pi}}\int_{\mathbb{R}} \exp(\mathrm{i}\,t\,(\mu-\mu') - \frac{1}{2}(\sigma^2+\sigma'^2+\frac{1}{\eta^2})t^2)\mathrm{d}t \\ &= \frac{1}{\eta\hat{\sigma}}\exp(\frac{-\hat{\mu}^2}{2\hat{\sigma}^2}), \qquad \text{(integral of a Gaussian function)}\end{aligned} \tag{16}$$

where $\hat{\mu} = \mu - \mu'$ and $\hat{\sigma} = \sqrt{\sigma^2 + \sigma'^2 + 1/\eta^2}$. Therefore, at univariate leaf nodes, it can be solved either analytically or with Monte-Carlo integration:

$$\int_{\mathbb{R}} \mathsf{L}(t)\overline{\mathsf{L}'(t)}\omega(t;\eta)\mathrm{d}t = \frac{1}{k}\sum_{j=1}^{k} \varphi_{\mathsf{L}}(t_j)\overline{\varphi_{\mathsf{L}'}(t_j)}, \tag{17}$$

where $\{t_1,\cdots,t_k\} \overset{\text{i.i.d.}}{\sim} \omega(t;\eta)$. With the above properties, the CFD between two compatible CCs can be calculated from bottom-up analytically and efficiently.