Isotropic Gaussian Processes on Finite Spaces of Graphs

Isotropic Gaussian Processes on Finite Spaces of Graphs

Viacheslav Borovitskiy[*1] **Mohammad Reza Karimi**[*1] **Vignesh Ram Somnath**[*1, 2] **Andreas Krause**[1]
[1] Learning & Adaptive Systems Group, Department of Computer Science, ETH Zürich, Switzerland
[2]IBM Research Zürich, Switzerland

Abstract

We propose a principled way to define Gaussian process priors on various sets of unweighted graphs: directed or undirected, with or without loops. We endow each of these sets with a geometric structure, inducing the notions of closeness and symmetries, by turning them into a vertex set of an appropriate metagraph. Building on this, we describe the class of priors that respect this structure and are analogous to the Euclidean isotropic processes, like squared exponential or Matérn. We propose an efficient computational technique for the ostensibly intractable problem of evaluating these priors' kernels, making such Gaussian processes usable within the usual toolboxes and downstream applications. We go further to consider sets of equivalence classes of unweighted graphs and define the appropriate versions of priors thereon. We prove a hardness result, showing that in this case, exact kernel computation cannot be performed efficiently. However, we propose a simple Monte Carlo approximation for handling moderately sized cases. Inspired by applications in chemistry, we illustrate the proposed techniques on a real molecular property prediction task in the small data regime.

1 Introduction

Gaussian processes provide a principled framework to assess and quantify uncertainty, making them useful in various applications, e.g., in optimization (Snoek et al., 2012), active & reinforcement learning (Deisenroth and Rasmussen, 2011; Krause and Guestrin, 2007).

Traditionally, Gaussian processes are applied to model functions $f : X \to \mathbb{R}$ where $X = \mathbb{R}^n$ is a Euclidean space. However, many applications require modeling functions on different domains X. The main ingredient needed for this is defining a natural Gaussian process prior on X. It should respect the geometric structure of X and, at the same time, be fairly general-purpose in its nature.

In the Euclidean case, applications often rely on *isotropic* priors, i.e., priors whose distribution is invariant with respect to translations and rotations, like squared exponential (RBF, Gaussian) or Matérn Gaussian processes.[1] When conditioning such a prior by a translated and rotated dataset, the resulting model is transformed accordingly.

Extending this notion of isotropy to non-Euclidean X has been a subject of recent work. For instance, Borovitskiy et al. (2021, 2020) and Azangulov et al. (2022, 2023) consider X that is a Riemannian manifold or a vertex set of an undirected graph, where the notion of isotropy is substituted with invariance to Riemannian isometries or graph automorphisms, respectively. Relying on Bochner's theorem-like constructions, they define appropriate priors and study (approximate) computational routines necessary for using them in various applications, e.g., in robotics (Jaquier et al., 2022) or wind speed modeling (Hutchinson et al., 2021).

Following this principled paradigm, and motivated by applications in natural sciences, e.g. in chemistry, in this work, we consider domains X that are sets of unweighted graphs on n vertices (directed or undirected, with or without loops). We endow these sets with appropriate geometric structure, turning them into *spaces*, and *derive* the generalized notion of isotropic Gaussian processes thereon. We obtain, as special cases, the analogs of the landmark Matérn and squared exponential Gaussian processes. This, however, leaves us with ostensibly intractable kernels which we *make tractable* by leveraging a finer structure of the setting.

We further consider a natural extension of the previous setting, whereby X is now a set of equivalence classes of graphs under some permutation-induced equivalence relation, e.g., the set of graph isomorphism classes. One re-

Proceedings of the 26[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2023, Valencia, Spain. PMLR: Volume 206.

[*]Equal contribution. Mail to: viacheslav.borovitskiy@gmail.com
Code available at: https://github.com/vsomnath/graph_space_gps. Mirrored at https://github.com/IBM/graph_space_gps.

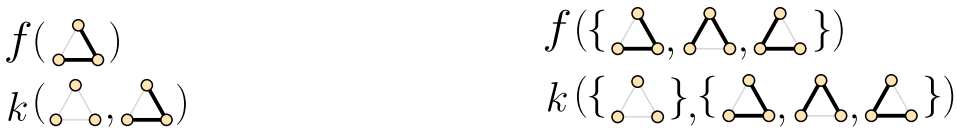


Figure 1: We study Gaussian processes f and their respective covariance kernels k in two settings. First, f that take *graphs* as inputs (left). Second, f that take *equivalence classes of graphs*, e.g., isomorphism classes of graphs, as inputs (right).

alistic use case of this arises when using graphs to encode molecules by associating nodes to atoms and edges to chemical bonds. Here, nodes corresponding to the same kind of atom are interchangeable while nodes corresponding to different types of atoms are not, calling for an equivalence class representation of a molecule, rather than for a graph representation. We illustrate the two different settings we study in Figure 1.

To handle this setting, we propose *projecting* the previously defined Gaussian process priors to make them into piecewise constant random functions on graph equivalence classes which we call *invariant versions*. We prove a hardness result that suggests that exact evaluation of kernels of the invariant versions cannot be done in any computationally efficient way, and suggest using straightforward Monte Carlo approximation to handle moderately sized problems.

To lend empirical support to our theoretical contributions, we evaluate the proposed methods on a real molecular property prediction task that mimics a typical application setting for Gaussian processes. We also consider a smaller subset of the same dataset where we can exactly evaluate the *projected* kernels, i.e. the kernels of invariant versions, further demonstrating the utility of such geometrically structured priors when learning from limited data.

1.1 Gaussian Processes

Gaussian processes (Rasmussen and Williams, 2006) are used as nonparametric probabilistic models for learning unknown functions. A Gaussian process $f \sim \mathrm{GP}(m, k)$ is a random function from some domain X to reals. Its distribution is determined by its *mean function* $m(x) = \mathbb{E}\, f(x)$ and its *covariance kernel* $k(x, x') = \mathrm{Cov}(f(x), f(x'))$, where the latter is necessarily *positive semidefinite*.

Given a zero-mean Gaussian process prior $f \sim \mathrm{GP}(0, k)$ and some data $x_1, y_1, \ldots, x_n, y_n$ with $x_i \in X, y_i \in \mathbb{R}$, one popular setting is to assume a Bayesian model $y_i = f(x_i) + \varepsilon_i$ with observations contaminated by i.i.d. noise $\varepsilon_i \sim \mathrm{N}(0, \sigma_\varepsilon^2)$. If we denote $\boldsymbol{y} = (y_1, \ldots, y_n)^\top$, this leads to posterior predictive $f \mid \boldsymbol{y} \sim \mathrm{GP}(\hat{m}, \hat{k})$ given by pathwise conditioning (Wilson et al., 2020; Wilson et al., 2021)

$$f \mid \boldsymbol{y}\,(\cdot) = f(\cdot) + \mathbf{K}_{\cdot \boldsymbol{x}}\big(\mathbf{K}_{\boldsymbol{xx}} + \sigma_\varepsilon^2 \mathbf{I}\big)^{-1}(\boldsymbol{y} - f(\boldsymbol{x})), \quad (1)$$

where $\boldsymbol{x}$ is defined similarly to $\boldsymbol{y}$, $\mathbf{I}$ is the identity matrix, $f(\boldsymbol{x}) = (f(x_1), \ldots, f(x_n))^\top$, and $\mathbf{K}_{\boldsymbol{ab}}$ is the matrix with elements $(\mathbf{K}_{\boldsymbol{ab}})_{ij} = k(a_i, b_j)$. Posterior moments $\hat{m}, \hat{k}$ are easily inferred to be

$$\hat{m}\,(\cdot) = \mathbf{K}_{\cdot \boldsymbol{x}}\big(\mathbf{K}_{\boldsymbol{xx}} + \sigma_\varepsilon^2 \mathbf{I}\big)^{-1}\boldsymbol{y}, \quad (2)$$

$$\hat{k}\,(\cdot, \cdot') = k(\cdot, \cdot') - \mathbf{K}_{\cdot \boldsymbol{x}}\big(\mathbf{K}_{\boldsymbol{xx}} + \sigma_\varepsilon^2 \mathbf{I}\big)^{-1}\mathbf{K}_{\boldsymbol{x} \cdot'}. \quad (3)$$

The posterior mean function $\hat{m}(x)$ evaluated at $x \in X$ is the prediction at x, while the posterior standard deviation $\hat{k}(x, x)^{1/2}$ therein represents the respective uncertainty.

In order to use Gaussian processes in downstream applications, one needs a suitable prior and an inference algorithm. For regression with Gaussian noise, the latter is given by Equations (1) to (3). In other settings, e.g., in classification, inference is not so straightforward, but is well-studied (Blei et al., 2017; Hensman et al., 2015). Crucially, inference algorithms transfer to new domains in a straightforward way if kernel pointwise evaluation and prior sampling are available thereon. We thus concentrate on building natural priors for Gaussian processes on various finite sets of graphs and their equivalence classes, which we proceed to discuss in the following section.

1.2 Finite Spaces of Graphs and Equivalence Classes

We study various sets of unweighted graphs on n nodes. Every such graph may be represented by its adjacency matrix or, after flattening to a vector, by an element of the set $\{0, 1\}^d$ with an appropriate d. For example, $d = n^2$ for the set of directed graphs with loops, which we denote by $\mathcal{DL}_n$, where $\mathcal{D}$ stands for *directed* and $\mathcal{L}$ for *loops*.

We also consider the sets of undirected graphs with loops (denoted by $\mathcal{UL}_n$, where $\mathcal{U}$ stands for *undirected*), directed graphs without loops (denoted by $\mathcal{D}_n$) and undirected graphs without loops (denoted by $\mathcal{U}_n$). Taking into account the structure of their adjacency matrices, these can be regarded as sets $\{0, 1\}^d$ with $d = n(n+1)/2$, $d = n(n-1)$ and $d = n(n-1)/2$, respectively. We turn these sets into *spaces* in Section 2, by endowing them with an appropriate geometric structure.

We also consider sets of equivalence classes of graphs. Extreme examples of these are isomorphism classes of graphs. To define these, consider the group S_n of permutations of a

[1]Note: ARD versions (Rasmussen and Williams, 2006, page 106) of Matérn and squared exponential Gaussian processes are *stationary* but not isotropic since they are not rotation invariant.

size n set.[2] A permutation $\sigma \in \mathrm{S}_n$ acts on a graph x with n nodes and adjacency matrix $\mathbf{A}_x$ returning the graph $y = \sigma(x)$ whose adjacency matrix $\mathbf{A}_y$ has rows and columns permuted by σ. Graphs x and y are called *isomorphic* (denoted by $x \cong y$) if and only if there exists $\sigma \in \mathrm{S}_n$ such that $\sigma(x) = y$. This defines an equivalence relation on any set $\mathcal{V}$ of graphs and thus defines the equivalence classes (called *graph isomorphism classes*): for $x \in \mathcal{V}$ its $\cong$-equivalence class is $\overline{x} = \{z \in \mathcal{V} : z \cong x\}$. The set of such equivalence classes we denote by $\mathcal{V}_{/\cong} = \{\overline{x} : x \in \mathcal{V}\}$.

We also consider sets of equivalence classes defined by general equivalence relations determined by various subgroups $H \subseteq \mathrm{S}_n$. Such a subgroup H defines the equivalence relation $\sim_H$ on any set $\mathcal{V}$ of graphs where $x \sim_H y$ if and only if there exists $\sigma \in H$ such that $\sigma(x) = y$. Obviously, $\sim_{\mathrm{S}_n}$ is equal to the relation $\cong$ considered above.

1.3 Previous Work and Contribution

Defining a zero-mean Gaussian process prior amounts to defining a kernel. Kernels on various sets of graphs have been under consideration for a long time, see the recent surveys by Nikolentzos et al. (2021) and Kriege et al. (2020). The focus in previous work, however, is usually shifted towards other settings, most notably the one where node and edge features are more significant than the graphs' topology. Moreover, the respective kernels are usually based on heuristics or aimed for a specific use case. To our best knowledge, principled general purpose kernels, like Matérn kernels on graph spaces, have not been considered so far.

Perhaps closest to our work is the piece in Kondor and Lafferty (2002) about the hypercube diffusion kernel. As we show later, this is the kernel of the squared exponential prior that we propose in Section 2. However, their kernel was studied without graph spaces in mind, and tractability was ensured using techniques that do not generalize to other isotropic kernels we study in Section 2.

Kernels on graph isomorphism classes were studied before as well, see, e.g., Shervashidze et al. (2011). Notably, Gärtner et al. (2003) prove that every strictly positive definite kernel of this sort cannot be computed exactly in an efficient way. Our no-go results of Section 3 are similar.

Our main contributions are the following: (1) we propose a principled way of defining Gaussian process priors on finite graph spaces, endowing graph sets with geometric structure, thus making them into *spaces*; (2) we make the respective ostensibly intractable kernels tractable; (3) we propose a principled way of projecting these priors on various sets of graph equivalence classes; (4) we prove a hardness result about the kernels between equivalence classes forbidding efficient *exact* computation and (5) we demonstrate the performance of proposed methods on a real molecular property prediction task in the small data regime.

[2]For a discussion on the topic of groups we refer the reader to Kondor (2008) and Robinson (2003).

2 Priors on Finite Spaces of Graphs

Let $\mathcal{V}$ be any of the sets $\mathcal{U}_n, \mathcal{UL}_n, \mathcal{D}_n, \mathcal{DL}_n$ of unweighted graphs on n nodes, identified with $\{0,1\}^d$ for the appropriate d (see Section 1.2 for definitions). Note that $|\mathcal{V}| = 2^d$.

Defining a reasonable Gaussian process prior on $\mathcal{V}$ requires endowing the set $\mathcal{V}$ with some sort of "geometric" structure, which — as a bare minimum — defines a notion of closeness. Arguably the most general way of encoding a geometric structure of a finite set is by making it into a vertex set of a graph. Moreover, principled and practical Gaussian processes on nodes of finite graphs were studied before (Borovitskiy et al., 2021; Kondor and Lafferty, 2002), meaning that we can build upon the previous work.

To avoid confusion, we will refer to the graph whose nodes are graphs as the *metagraph* and denote it by $\mathcal{G}$. We propose the following natural structure for $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. It is an unweighted undirected graph such that $(x, y) \in \mathcal{E}$ if and only if the graph y can be obtained from the graph x by adding or deleting a single edge. The notion of closeness defined by this structure corresponds to the Hamming distance, the number of differing bits in the representations of the graphs $x, y \in \mathcal{V}$ as vectors in the set $\{0,1\}^d$.

This structure, however, also defines some notion of symmetries: the $\mathcal{G}$'s own group of automorphisms $\mathrm{Aut}(\mathcal{G})$ that consists of all bijective maps $\phi : \mathcal{V} \to \mathcal{V}$ such that $(\phi(x), \phi(y)) \in \mathcal{E} \iff (x, y) \in \mathcal{E}$ for all $x, y \in \mathcal{V}$.

It is natural to ask that the distribution of a Gaussian process prior on $\mathcal{V}$ is invariant with respect to symmetries from $\mathrm{Aut}(\mathcal{G})$, similar to how the standard Euclidean squared exponential and Matérn priors are invariant with respect to translations and rotations, i.e., to the Euclidean isometries.

Definition. We call $f \sim \mathrm{GP}(0, k)$ on $\mathcal{V}$ *isotropic* if and only if for all $\phi \in \mathrm{Aut}(\mathcal{G})$ and finite sets $x_1, \ldots, x_n \in \mathcal{V}$

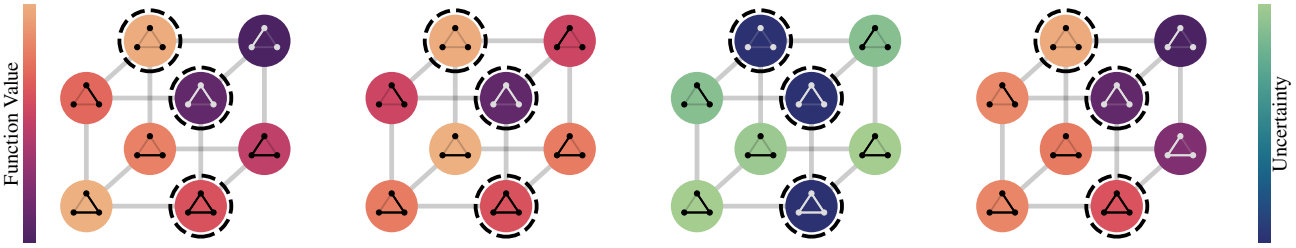
$$f(x_1, \ldots, x_n) \overset{\mathrm{d}}{=} f(\phi(x_1), \ldots, \phi(x_n)) \tag{4}$$

where $\overset{\mathrm{d}}{=}$ denotes equality in distribution. Or, equivalently,

$$k(\phi(x), \phi(y)) = k(x, y) \tag{5}$$

for all $\phi \in \mathrm{Aut}(\mathcal{G})$ and for all pairs $x, y \in \mathcal{V}$. We call the kernel of an isotropic process an *isotropic kernel*.

As we will show in Section 2.1, this class includes graph Matérn Gaussian processes $\mathrm{GP}(0, k_{\nu,\kappa,\sigma^2})$ on $\mathcal{G}$ in the sense of Borovitskiy et al. (2021), including, as special case for $\nu = \infty$, the Gaussian processes with diffusion (heat, squared exponential) kernels on hypercube graphs studied by Kondor and Lafferty (2002). If we define $\mathbf{\Delta}$ to



(a) Ground Truth (b) Prediction (c) Uncertainty (d) Posterior sample

Figure 2: A toy regression problem on the space $\mathcal{U}_3$ of undirected graphs with 3 nodes identified with the 3D cube graph. On these plots, color represents value of the corresponding function and training locations are marked by dashed outline.

be the graph Laplacian of $\mathcal{G}$ and (λ_j, f_j) to be its eigenpairs, where $\{f_j\}$ form an orthonormal basis in the set $L^2(\mathcal{V}) = \mathbb{R}^{|\mathcal{V}|}$ of functions on the finite set $\mathcal{V}$, these two kernels are given by

$$k_{\nu,\kappa,\sigma^2}(x,y) = \sum_{j=1}^{|\mathcal{V}|} \Phi_{\nu,\kappa,\sigma^2}(\lambda_j) f_j(x) f_j(y), \tag{6}$$

$$\Phi_{\nu,\kappa,\sigma^2}(\lambda) = \begin{cases} \frac{\sigma^2}{C_{\infty,\kappa}} e^{-\frac{\kappa^2}{2}\lambda}, & \text{for } \nu = \infty, \\ \frac{\sigma^2}{C_{\nu,\kappa}} \left(\frac{2\nu}{\kappa^2} + \lambda\right)^{-\nu} & \text{for } \nu < \infty. \end{cases} \tag{7}$$

In Figure 2 we illustrate a Gaussian process regression with a Matérn kernel ($\nu = 3.5$) on the set $\mathcal{V} = \mathcal{U}_3$ that will be made possible thanks to the developments of this section.

Clearly, any function $\Phi : \mathbb{R} \to [0, \infty)$ placed instead of $\Phi_{\nu,\kappa,\sigma^2}$ defines a valid covariance kernel. An implicit degree of freedom here is the definition of $\mathbf{\Delta}$. If $\mathbf{A}$ is the adjacency matrix of $\mathcal{G}$ and $\mathbf{D}$ is its diagonal degree matrix with $\mathbf{D}_{ii} = \sum_{j=1}^{|\mathcal{V}|} \mathbf{A}_{ij}$, there are three popular definitions of $\mathbf{\Delta}$: (1) the usual Laplacian $\mathbf{\Delta} = \mathbf{D} - \mathbf{A}$, (2) the random walk normalized Laplacian $\mathbf{\Delta}_{rw} = \mathbf{I} - \mathbf{D}^{-1}\mathbf{A}$ and (3) the symmetric normalized Laplacian $\mathbf{\Delta}_{sym} = \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$. The class of all kernels given in form of Equation (6) with any $\Phi : \mathbb{R} \to [0, \infty)$ we term Φ-*kernels*.[3]

Notice that although absolutely explicit, expressions like Equation (6) are ostensibly intractable: they entail (1) solving the eigenproblem for the $2^d \times 2^d$-sized Laplacian matrix and (2) summing up the 2^d terms together. To come with an efficient computational algorithm and to better understand the properties of these and similar kernels we proceed to study the class of isotropic processes on $\mathcal{V}$.

2.1 The Class of Isotropic Gaussian Processes

To characterize the class of isotropic processes on $\mathcal{V}$ we need to introduce a few additional notions. First, we define the group $\mathbb{Z}_2^m$ to be the set $\{0,1\}^m$ endowed with the bitwise XOR operation, which we denote by $\dot{+}$. Obviously, this turns the set $\mathcal{V}$ itself into the group $\mathbb{Z}_2^d$. Second, for an element σ of the group of permutations S_d of a d-sized set define $\sigma(x) = (x_{\sigma(1)}, \ldots, x_{\sigma(d)})$ where x is a graph regarded as an element of $\{0,1\}^d$. With this, we have the following characterization of isotropic processes.

Theorem 1. *A Gaussian process $f \sim \mathrm{GP}(0, k)$ on $\mathcal{V}$ is isotropic if and only if for all $x, y, z \in \mathcal{V} = \mathbb{Z}_2^d$ and $\sigma \in \mathrm{S}_d$*

$$(i): \quad k(x \dot{+} z, y \dot{+} z) = k(x, y),$$
$$(ii): \quad k(\sigma(x), \sigma(y)) = k(x, y).$$

Proof. This is a corollary of describing $\mathrm{Aut}(\mathcal{G})$ as the semidirect product $\mathbb{Z}_2^d \rtimes \mathrm{S}_d$ — see the definition, relevant discussion and the detailed proof in Appendix A. □

This characterization mirrors the description of isotropic Gaussian processes on $\mathbb{R}^d$ as having kernels invariant to all translations, similar to (i), and to all rotations, similar to (ii).[4] Moreover, this characterization may be leveraged to obtain an explicit description of the isotropic kernels class.

Theorem 2. *$f \sim \mathrm{GP}(0, k)$ is isotropic on $\mathcal{V}$ if and only if*

$$k(x,y) = \sum_{j=0}^{d} \alpha_j \sum_{T \subseteq \{1,\ldots,d\}, |T|=j} w_T(x) w_T(y), \quad \alpha_j \geq 0, \tag{8}$$

where $w_T(x) = (-1)^{\sum_{t \in T} x_t}$ are the Walsh functions — the analogs of complex exponentials in the Fourier analysis of Boolean functions (O'Donnell, 2014).

Proof. From (i) it is possible to deduce that

$$k(x,y) = \sum_{T \subseteq \{1,\ldots,d\}} \alpha_T w_T(x) w_T(y), \quad \alpha_T \geq 0. \tag{9}$$

Then, (ii) yields $\alpha_T = \alpha_{T'}$ for all sets T, T' of the same size. See the detailed proof in Appendix A. □

[3] For our specific graph $\mathcal{G}$ we have $\mathbf{\Delta}_{rw} = \mathbf{\Delta}_{sym} = \mathbf{\Delta}/d$, hence f_j are always the same and λ_j only differ by a constant factor of d, defining the same class of Φ-kernels.

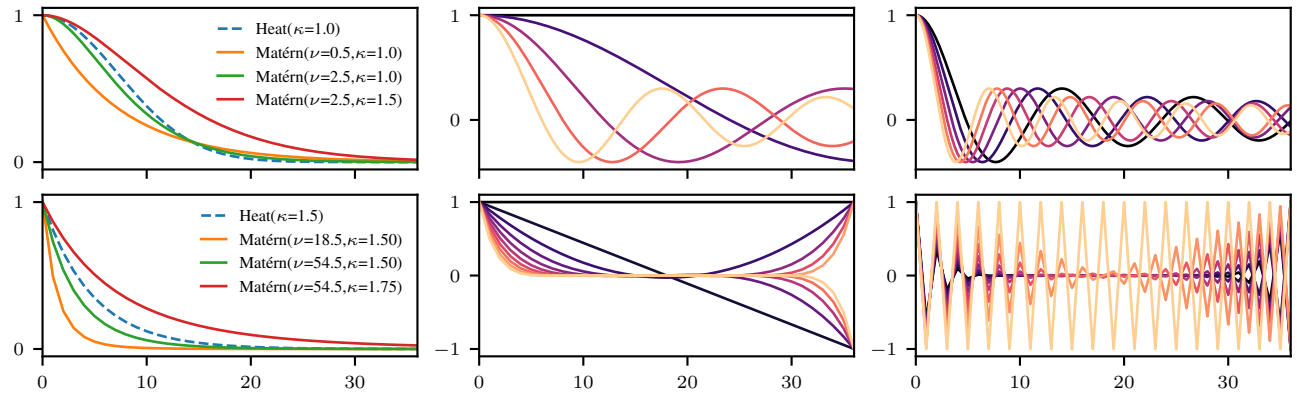


Figure 3: Isotropic kernels (left) and their low (center) and high (right) frequency components, as functions of distance, in the classical Euclidean case, for $\mathbb{R}^2$ (top row), and in the graph space case, on $\mathcal{DL}_6$ (bottom row). The basis functions on $\mathcal{DL}_6$ are $G_{36,j,m}$ as functions of m for various j. The basis functions on $\mathbb{R}^2$ arise from Hankel transform (Bracewell, 2000) and coincide with $J_0(2\pi qr)$ as functions of r for various q, where J denotes a Bessel function of the first kind.

On the other hand, we can prove that the class of Φ-kernels has exactly the same form.

Theorem 3. *The set $\{w_T(\cdot)\}_{T\subseteq\{1,\dots,d\}}$ is an orthonormal basis of $L^2(\mathcal{V})$ consisting of eigenfunctions of any of the Laplacians $\boldsymbol{\Delta}, \boldsymbol{\Delta}_{rw}, \boldsymbol{\Delta}_{sym}$ on $\mathcal{G}$ with eigenvalues*

$$\lambda_T^{\boldsymbol{\Delta}} = 2|T| \qquad \lambda_T^{\boldsymbol{\Delta}_{rw}} = \lambda_T^{\boldsymbol{\Delta}_{sym}} = 2|T|/d \tag{10}$$

Proof. The key is to recognize $\mathcal{G}$ as the *Cayley graph* of the group $\mathbb{Z}_2^d$. See the detailed proof in Appendix A. □

Corollary. *The class of isotropic kernels on $\mathcal{V}$ and the class of Φ-kernels on $\mathcal{G}$ coincide.*

2.2 Efficient Computation of Isotropic Kernels

Theorem 3 relieves us of the need to solve the eigenproblem for $\boldsymbol{\Delta}$ numerically. Although this saves us an immense number of $O(2^{3d})$ computational operations, computing $k(x, y)$ still entails summing up 2^d terms — an impossible problem for even a moderate d. For general graphs, Borovitskiy et al. (2021) recommend approximating $k(x, y)$ by truncating the sum in Equation (6). Below we show that a much more efficient solution exists.

Denote the inner sum in Equation (8) by

$$G_{d,j}(x, y) = \sum_{T\subseteq\{1,\dots,d\},|T|=j} w_T(x)w_T(y). \tag{11}$$

For $x \in \{0, 1\}^d$ define $|x| = \sum_{i=1}^d x_d$. Then we have the following characterization of $G_{d,j}(x, y)$.

[4]In fact, $\mathbb{Z}_2^d$ and S_d may be regarded as the groups of translations and rotations related to vector spaces over the *field of characteristic* 2 in place of the field $\mathbb{R}$, as discussed in Appendix A.

Theorem 4. *$G_{d,j}(x, y) = G_{d,j,m}$ where $m = |x \dot{+} y|$ is the Hamming distance between graphs x and y. Moreover,*

$$G_{d,j,m} = G_{d-1,j,m-1} - G_{d-1,j-1,m-1} \tag{12}$$

with $G_{1,j,m} = (-1)^m$, $G_{d,0,m} = 1$ and $G_{d,j,0} = \binom{d}{j}$.

Proof. See Appendix A. □

This means that $G_{d,j,m}$ may be computed by a simple dynamical programming procedure. Moreover, caching $G_{d,j,m}$ makes kernel evaluation for arbitrary Φ take only $O(d)$ computational operations. This can be further reduced to $O(1)$ by truncating $G_{d,j,m}$.[5]. This is very important for optimizing Gaussian process hyperparameters like length scale of Matérn kernels, when the kernels are evaluated with different functions Φ each iteration. We illustrate the graph and the Euclidean isotropic kernels on Figure 3.

Interestingly, by virtue of Equation (48) in Appendix A, functions $G_{d,j,m}$ coincide with the Kravchuk polynomials defined, e.g., in MacWilliams and Sloane (1977).

As a byproduct of sorts, we established that for an isotropic Gaussian process $\mathrm{GP}(0, k)$ we have

$$k(x, y) = \Bbbk(|x \dot{+} y|) \quad \text{for some } \Bbbk : \{1, .., d\} \to \mathbb{R} \tag{13}$$

showing that isotropic kernels do indeed respect the notion of closeness given by Hamming distance.

What is more, since for $x \in \{0, 1\}^d$ we have $|x| = \|x\|^2$, we see that for any $g : [0, \infty) \to [0, \infty)$ such that $\tilde{k}(x, y) = g(\|x - y\|^2)$ is positive semidefinite for $x, y \in \mathbb{R}^d$, its restriction $k(x, y) = g(|x \dot{+} y|)$ will be positive semidefinite and isotropic for $x, y \in \mathcal{V} = \{0, 1\}^d$.

[5]This truncation will implicitly incorporate an orders of magnitude larger number of Walsh function terms than the naïve one.

Finally, since the heat kernel on the direct product of graphs is the product of their heat kernels, by representing $\mathcal{G}$ as the iterated product of the two-vertex complete graphs, one can infer (Kondor and Lafferty, 2002) that

$$k_{\infty,\kappa,\sigma^2}(x,y) = \sigma^2 \tanh(\kappa^2/2)^{|x \dot{+} y|} \tag{14}$$

which coincides with the Euclidean squared exponential

$$\sigma^2 \exp\Big(-\frac{\|x-y\|^2}{2\kappa'}\Big) \text{ with } \kappa' = \frac{1}{2\log(\tanh(\kappa^2/2))}. \tag{15}$$

2.3 Sampling from Isotropic Gaussian Processes

Consider an isotropic Gaussian process $f \sim \mathrm{GP}(0,k)$. Since we are able to evaluate k efficiently, given any collection $x_1,\dots,x_m \in \mathcal{V}$ denoted by $\boldsymbol{x}$, sampling $f(\boldsymbol{x})$ may be performed by

$$f(\boldsymbol{x}) = \mathbf{K}_{\boldsymbol{x}\boldsymbol{x}}^{1/2}\boldsymbol{\varepsilon} \qquad\qquad \boldsymbol{\varepsilon} \sim \mathrm{N}(0,\mathbf{I}). \tag{16}$$

Computing $\mathbf{K}_{\boldsymbol{x}\boldsymbol{x}}^{1/2}$, however, requires $O(m^3)$ computational operations and is thus inefficient for larger m.

Let us examine alternatives for the setting at hand. Consider $k(x,y) = \sum_{j=0}^{d} \alpha_j G_{d,j}(x,y)$ with $\alpha_j \geq 0$ sorted in descending order. As by Borovitskiy et al. (2021), one alternative is to use the approximation

$$f(x) \approx \sum_{j=0}^{J} \sqrt{\alpha_j} \sum_{T\subseteq\{1,\dots,d\},|T|=j} \varepsilon_T w_T(x), \quad \varepsilon_T \overset{\text{i.i.d.}}{\sim} \mathrm{N}(0,1). \tag{17}$$

This has the advantage of defining a sample at all $x \in \mathcal{V}$ at once, not on a pre-specified small set $x_1,\dots,x_m \in \mathcal{V}$ as before, but with the downside of being approximate. The number of terms in the inner sum grows roughly like d^j, thus Equation (17) can be realistically evaluated only for small values of J. If using larger values of J is desirable, one approach, similar to the generalized random phase Fourier features of Azangulov et al. (2022) is

$$f(x) \approx \sum_{j=0}^{J} \frac{\sqrt{\alpha_j}}{\sqrt{L}} \sum_{l=1}^{L} \varepsilon_{j,l} G_{d,j}(x,u_l), \tag{18}$$

$$\varepsilon_{j,l} \overset{\text{i.i.d.}}{\sim} \mathrm{N}(0,1), \qquad u_l \overset{\text{i.i.d.}}{\sim} \mathrm{U}(\mathcal{V}). \tag{19}$$

where $\mathrm{U}(\mathcal{V})$ denotes the uniform distribution over the finite set $\mathcal{V}$. This way one can safely take $J = d$, but one also needs to choose the value of L.

These sampling techniques might be useful to perform non-conjugate learning via doubly stochastic variational inference as briefly reviewed in the relevant setting by Borovitskiy et al. (2021), to evaluate various Bayesian optimization acquisition functions or for visualization purposes etc.

3 Priors on Spaces of Equivalence Classes

Here we define and study natural Gaussian process priors on sets of equivalence classes of graphs. Recall that we start with some finite graph set $\mathcal{V}$ and a subgroup $H \subseteq \mathrm{S}_n$ of the group S_n of permutations of vertices. This subgroup defines an equivalence relation $\sim_H$ on $\mathcal{V}$ and the set of equivalence classes of graphs $\mathcal{V}_{/\sim_H}$. For example, if $H = \mathrm{S}_n$, then $\sim_H$ is the graph isomorphism relation. For convenience, we will write $\mathcal{V}_{/H}$ as a shorthand for $\mathcal{V}_{/\sim_H}$. We refer the reader to Section 1.2 for a more detailed discussion on equivalence classes.

Our strategy to build a prior on $\mathcal{V}_{/H}$ is to take a prior on $\mathcal{V}$ that is provided to us by Section 2, and *project* it, making it constant on each of the equivalence classes, i.e. such that $f(x) = f(y)$ for all $x \sim_H y$.

Consider the space $L^2(\mathcal{V})$ of functions on $\mathcal{V}$ and the operator $\mathrm{Pr} : L^2(\mathcal{V}) \to L^2(\mathcal{V})$ given by

$$(\mathrm{Pr}\, f)(x) = \frac{1}{|H|}\sum_{\sigma\in H} f(\sigma(x)) \tag{20}$$

Denote by W the subspace of $L^2(\mathcal{V})$ consisting of functions that are constant on each of the equivalence classes $\overline{x}$, $x \in \mathcal{V}$. We prove in Appendix B that the operator Pr is an orthogonal projector onto the $|\mathcal{V}_{/H}|$-dimensional space W.

Now we define the prior on $\mathcal{V}_{/H}$ corresponding to a prior on $\mathcal{V}$, which we call its *H-invariant version*.

Definition. Consider a Gaussian process $f \sim \mathrm{GP}(0,k)$ on $\mathcal{V}$. We call the Gaussian process

$$f_{/H}(x) = (\mathrm{Pr}\, f)(x) \quad \text{with kernel} \tag{21}$$

$$k_{/H}(x,y) = \frac{1}{|H|^2}\sum_{\sigma_1\in H}\sum_{\sigma_2\in H} k(\sigma_1(x),\sigma_2(y)) \tag{22}$$

the *H-invariant* version of f and the kernel $k_{/H}$ the *H-invariant* version of k.

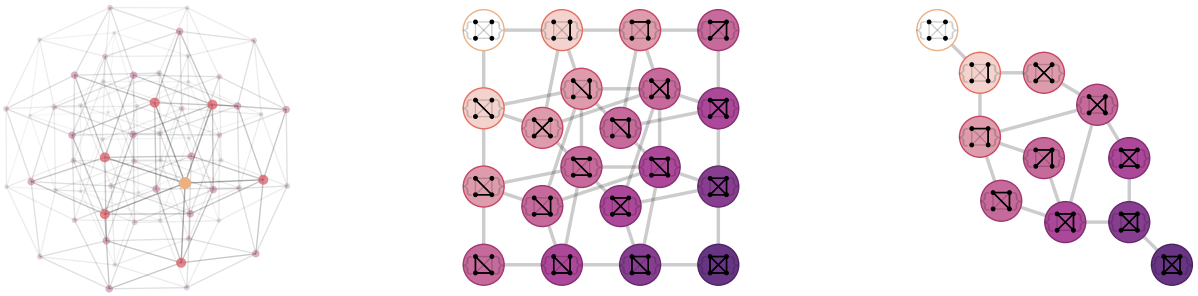
The term is justified because, as it is easy to see, for all $\sigma,\sigma_1,\sigma_2 \in H$ and all $x,y \in \mathcal{V}$ we have

$$f_{/H}(\sigma(x)) = f_{/H}(x), \tag{23}$$

$$k_{/H}(\sigma_1(x),\sigma_2(y)) = k_{/H}(x,y), \tag{24}$$

meaning that $f_{/H}$ and $k_{/H}$ are constant when restricted onto each (pair of) equivalence classes.

Interestingly, if we regard H-invariant versions as functions of equivalence classes (i.e., on the set $\mathcal{V}_{/H}$) rather than functions of graphs (i.e., on the set $\mathcal{V}$), they may also be seen as arising from certain metagraphs, more concretely from appropriately weighted *quotient graphs* (Cozzo et al., 2018). Specifically, let us define the graph $\mathcal{G}_{/H} = (\mathcal{V}_{/H}, \mathcal{E}_{/H})$ such that the weight of an edge $(\overline{x},\overline{y})$ is exactly the cardinality of the set $\{(x',y') \in \mathcal{E} : x' \in \overline{x}, y' \in \overline{y}\}$. Then, we have following.



(a) Hypercube graph (b) Quotient graph under $\sim_H$ (c) Quotient graph under $\sim_{S_4}$, i.e. $\cong$

Figure 4: Matérn kernel with $\nu = 8.5$ on the hypercube graph $\mathcal{V} = \mathcal{U}_4$ and on the quotient graphs corresponding to $\mathcal{V}_{/H}$ and $\mathcal{V}_{/\mathrm{S}_4}$, where $H = \mathrm{S}_3 \times \mathrm{S}_1 \subseteq \mathrm{S}_4$, i.e. $x \sim_H y$ if and only if the first three vertices of x may be permuted to make x into y. Color of a node represents kernel's value between this node and the (equivalence class of) the empty graph.

Theorem 5. *Consider the class of Φ-kernels induced by the symmetric normalized Laplacian, and take a Gaussian process $f \sim \mathrm{GP}(0, k)$ on $\mathcal{V}$ with a Φ-kernel given by a function Φ. Its H-invariant version $f_{/H} \sim \mathrm{GP}(0, k_{/H})$ has kernel $k_{/H}(x, y) = \psi(\overline{x})\psi(\overline{y})k_\Phi(\overline{x}, \overline{y})$ where k_Φ is the Φ-kernel on the quotient graph $\mathcal{G}_{/H}$ with the same function Φ. Moreover, $\psi(\overline{x}) = |\overline{x}|^{-1/2}$.*

Proof. See Appendix B. □

When reinterpreting a function from $W \subseteq L^2(\mathcal{V})$ as a function in $L^2(\mathcal{V}_{/H})$, the norm is not preserved. This explains why the factor $\psi(\cdot)$ appears in the theorem above.

We illustrate Matérn kernels on the graph $\mathcal{G}$ for $\mathcal{V} = \mathcal{U}_4$, the quotient graph corresponding to $H = \mathrm{S}_3 \times \mathrm{S}_1 \subseteq \mathrm{S}_4$ and the quotient graph corresponding to the graph isomorphism relation $\cong$ in Figure 4.

After we introduced a way to build priors on $\mathcal{V}_{/H}$, we turn to the associated computational routines.

3.1 Kernel Computation and Sampling

We start by proving a hardness result suggesting it is impossible to compute $k_{/H}$ exactly in an efficient way.

Theorem 6. *Assume k is a Φ-kernel on $\mathcal{G}$ for a strictly positive Φ. Exactly computing $k_{/H}$ at three pairs of inputs $(x, y), (x, x)$ and (y, y) is at least as hard as checking whether $x \sim_H y$. In particular, if $H = \mathrm{S}_n$, it is at least as hard as resolving the graph isomorphism problem.*[6]

Proof. We prove that $x \sim_H y$ is equivalent to the condition

$$k(x, y) = \frac{k(x, x) + k(y, y)}{2}. \tag{25}$$

See Appendix B for details. □

[6]Whether or not this may be done in polynomial time is currently considered an open problem.

This effectively forbids exact evaluation of $k_{/H}$ for larger values of n. For small n, e.g., $n \leq 10$, this can be done simply by definition. For moderately larger values of n, one may consider using the Monte Carlo approximation

$$k_{/H}(x, y) \approx \frac{1}{|S|^2} \sum_{\sigma_1 \in S} \sum_{\sigma_2 \in S} k(\sigma_1(x), \sigma_2(y)), \tag{26}$$

$$S \subseteq H, \qquad S \ni \sigma \overset{\text{i.i.d.}}{\sim} \mathrm{U}(H), \tag{27}$$

where $\mathrm{U}(H)$ denotes uniform distribution over the set H.

Remark. The right-hand side of Equation (26) is necessarily positive semidefinite. One can use the isotropy of k to introduce a different approximation, $k_{/H}(x, y) = \frac{1}{|S|} \sum_{\sigma \in S} k(\sigma(x), y)$ and hope for better convergence as

$$k_{/H}(x, y) = \frac{1}{|H|^2} \sum_{\sigma_1, \sigma_2 \in H} k(\sigma_2^{-1}\sigma_1(x), y) \tag{28}$$

$$= \frac{1}{|H|} \sum_{\sigma \in H} k(\sigma(x), y). \tag{29}$$

However, this approximation can easily fail to be positive semidefinite (even symmetric!), hence the approximation in Equation (26) will usually be preferable in practice.

Finally, we discuss sampling $f_{/H} \sim \mathrm{GP}(0, k_{/H})$. Mirroring the approximation in Equation (26), to draw a sample $\hat{f}_{/H}$ we suggest drawing a sample $\hat{f} \sim \mathrm{GP}(0, k)$ and putting

$$\hat{f}_{/H}(x) = \sum_{\sigma \in S} \frac{1}{|S|} \hat{f}(\sigma(x)). \tag{30}$$

It is trivial to check that if $\hat{f}$ is an exact sample from $\mathrm{GP}(0, k_{/H})$, then the covariance of the right-hand side will exactly coincide with the right-hand side of Equation (26).

4 Experimental Setup

Inspired by applications in chemistry, we evaluate our proposed models on a real molecular property prediction task.

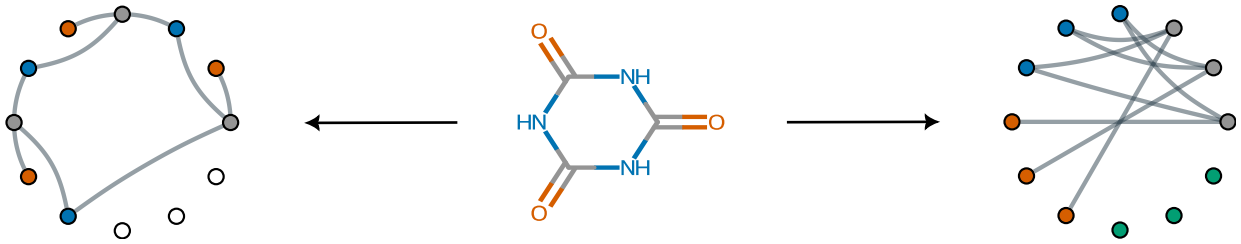


Figure 5: Transforming molecules into graphs. Given a molecule (middle), we construct a graph by assigning certain atoms to arbitrary nodes (left), and by assigning atoms to certain pre-specified groups of nodes (right). Nodes (and node groups) carry the same color as the corresponding atoms.

To match typical application settings for Gaussian processes, we consider a small dataset. Specifically, we utilize the FreeSolv (Mobley and Guthrie, 2014) dataset provided as part of the MoleculeNet benchmark (Wu et al., 2018). It consists of 642 molecules with experimentally measured hydration free energy values. After removing invalid molecules, we utilize a 80/20 train/test split to obtain 510 and 128 examples for training and testing, respectively.

We also consider a subset of FreeSolv (FreeSolv-S), allowing upto four atom types (carbon, nitrogen, oxygen and chlorine), with a maximum of 3 atoms per atom type. This results in 52 training and 13 text examples, and allows us to test exact projected Gaussian processes and demonstrate reasonable performance with even more limited data.

To construct graphs from molecules, we assign atoms to nodes, and corresponding chemical bonds to edges. We adopt two strategies - a) assigning atoms to arbitrary nodes in the graph (GRAPH-B in Table 1, left on Figure 5), and b) assigning certain types of atoms to certain pre-specified groups of vertices thus aligning different data samples better (GRAPH-A in Table 1, A stands for aligned, right on Figure 5). Both strategies are illustrated in Figure 5.

Table 1: Molecule property prediction performance. RMSE is reported on the original data scale, with standard deviation of 3.89 on FreeSolv and 3.83 on FreeSolv-S.

Method	**FreeSolv**		**FreeSolv-S**	
	Log Lik.	RMSE	Log Lik.	RMSE
NAIVE	—	0.32	—	0.99 ± 0.25
LINEAR	-154.37	0.27	-19.44 ± 5.19	1.06 ± 0.26
GRAPH-B				
- Heat	-151.28	0.26	-19.42 ± 5.41	1.05 ± 0.26
- Matérn	-151.19	0.26	-19.65 ± 5.11	1.05 ± 0.25
GRAPH-A				
- Heat	-77.55	0.17	-10.65 ± 4.59	0.67 ± 0.31
- Matérn	-77.54	0.17	-10.69 ± 4.45	0.67 ± 0.30
PROJECTED				
- Heat	—	—	-3.88 ± 3.40	0.50 ± 0.20
- Matérn	—	—	-4.26 ± 3.37	0.49 ± 0.20

Varying number of nodes. To handle graphs with varying number of nodes, we use Gaussian processes on the metagraph corresponding to the maximal number of nodes throughout the dataset. All graphs are than "padded", if necessary, by additional disconnected nodes. These can be seen at the bottom right of the graphs presented on Figure 5.

We compare our proposed methods to the NAIVE baseline that outputs the mean over the train set and a LINEAR kernel based Gaussian process. On the FreeSolv-S dataset, we also evaluate the projected kernels (PROJECTED), where the group of H is such that $x \sim_H y$ if and only if x can be made into y by permuting only atoms of the same type (carbons with carbon, oxygens with oxygen etc.). Results are displayed in Table 1, with more details in Appendix C, and discussed, along with the theoretical contributions, in Section 5. For the Freesolv-S dataset, we generate 10 random splits and report the mean and standard deviation of the metrics across these splits.

5 Discussion and Conclusion

The ease of use of the graph heat kernel, granted by a closed form formula in Equation (14), and its fair performance observed in experiments make it a good baseline kernel to consider in the absence of other structure. The theory of this paper places it onto a firmer foundation, showing how it arises from endowing graph sets with a natural geometry.

Graph Matérn kernels and the heat kernel may slightly outperform each other depending on the setting and the metric. It is therefore interesting to characterize settings where and to what extent one kernel is better than the other. Explicit spectral representations we derived both for finite-ν Matérn and heat kernels may be key to this, as they often appear in providing regression convergence rates (Kanagawa et al., 2018). These spectral representations may also be relevant for inferring regret bounds in Bayesian optimization (Srinivas et al., 2010) or related active learning techniques.

Our results suggest that projected graph Gaussian processes may heavily outperform simple graph Gaussian processes in problems possessing inherent invariances, where performance of the simple models is hindered by varied graph alignments — though appropriate data preprocess-

ing can already make a difference. Since exact computation of projected kernels is precluded by Theorem 6, further research into building efficient approximations thereof — which are not forbidden by the result — is needed.

Finally, we believe that the geometric framework treating graph spaces as vertex sets of appropriate metagraphs is quite general and a promising direction for further development. For example, we anticipate it to easily extend to certain settings with discretely labeled edges, where the metagraph can be chosen in such a way that isotropic processes are described in terms of other simple bases, similar to how they are described by the Walsh system in this paper.

To conclude, we hope that the proposed framework and our current developments set up the scene and guide further research interest for Gaussian process based modeling on spaces of graphs and their corresponding prospective applications like Bayesian optimization, active learning, etc.

Acknowledgments

The authors are grateful to Prof. Risi Kondor (The University of Chicago) and Dr. Konstantin Golubev (Google) for fruitful discussions. This work was supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme grant aggreement No 815943, the NCCR Catalysis (grant number 180544), a National Centres of Competence in Research funded by the Swiss National Science Foundation, and an ETH Zürich Postdoctoral Fellowship to VB. VRS acknowledges funding from the European Union's Horizon 2020 research and innovation programme 826121.

References

I. Azangulov, A. Smolensky, A. Terenin, and V. Borovitskiy. Stationary Kernels and Gaussian Processes on Lie Groups and their Homogeneous Spaces I: the compact case. *arXiv preprint arXiv:2208.14960*, 2022. Cited on pages 1, 6.

I. Azangulov, A. Smolensky, A. Terenin, and V. Borovitskiy. Stationary Kernels and Gaussian Processes on Lie Groups and their Homogeneous Spaces II: non-compact symmetric spaces. *arXiv preprint arXiv:2301.13088*, 2023. Cited on page 1.

D. M. Blei, A. Kucukelbir, and J. D. McAuliffe. Variational Inference: A Review for Statisticians. *Journal of the American statistical Association*, 112(518):859–877, 2017. Cited on page 2.

V. Borovitskiy, I. Azangulov, A. Terenin, P. Mostowsky, M. Deisenroth, and N. Durrande. Matérn Gaussian processes on graphs. In *International Conference on Artificial Intelligence and Statistics*, 2021. Cited on pages 1, 3, 5, 6.

V. Borovitskiy, A. Terenin, P. Mostowsky, and M. P. Deisenroth. Matérn Gaussian processes on Riemannian manifolds. In *Advances in Neural Information Processing Systems*, 2020. Cited on page 1.

R. N. Bracewell. *The Fourier Transform and Its Applications*. McGraw Hill, 2000. Cited on page 5.

E. Cozzo, G. F. De Arruda, F. A. Rodrigues, and Y. Moreno. *Multiplex Networks: Basic Formalism and Structural Properties*. Springer, 2018. Cited on pages 6, 15.

M. Deisenroth and C. E. Rasmussen. PILCO: A model-based and data-efficient approach to policy search. In *International Conference on Machine Learning*, 2011. Cited on page 1.

D. S. Dummit and R. M. Foote. *Abstract Algebra*, volume 3. Wiley Hoboken, 2004. Cited on page 11.

T. Gärtner, P. Flach, and S. Wrobel. On Graph Kernels: Hardness Results and Efficient Alternatives. In *Learning Theory and Kernel Machines*, pages 129–143. Springer, 2003. Cited on pages 3, 17.

C. Godsil and G. F. Royle. *Algebraic Graph Theory*, volume 207. Springer Science & Business Media, 2001. Cited on page 11.

J. Hensman, A. Matthews, and Z. Ghahramani. Scalable variational Gaussian process classification. In *Artificial Intelligence and Statistics*, 2015. Cited on page 2.

M. Hutchinson, A. Terenin, V. Borovitskiy, S. Takao, Y. Teh, and M. Deisenroth. Vector-valued Gaussian Processes on Riemannian Manifolds via Gauge Independent Projected Kernels. In *Advances in Neural Information Processing Systems*, volume 34, 2021. Cited on page 1.

N. Jaquier, V. Borovitskiy, A. Smolensky, A. Terenin, T. Asfour, and L. Rozo. Geometry-aware Bayesian Optimization in Robotics using Riemannian Matérn Kernels. In *Conference on Robot Learning*, 2022. Cited on page 1.

M. Kanagawa, P. Hennig, D. Sejdinovic, and B. K. Sriperumbudur. Gaussian Processes and Kernel Methods: A Review on Connections and Equivalences. *arXiv preprint arXiv:1807.02582*, 2018. Cited on page 8.

A. Kerber. *Representations of Permutation Groups I: Representations of Wreath Products and Applications to the Representation Theory of Symmetric and Alternating Groups*. Springer, 2006. Cited on page 11.

R. I. Kondor. *Group theoretical methods in machine learning*. Columbia University, 2008. Cited on page 3.

R. I. Kondor and J. Lafferty. Diffusion kernels on graphs and other discrete structures. In *International Conference on Machine Learning*, 2002. Cited on pages 3, 6, 11.

A. Krause and C. Guestrin. Nonmyopic active learning of Gaussian processes: an exploration-exploitation approach. In *International Conference on Machine Learning*, 2007. Cited on page 1.

N. M. Kriege, F. D. Johansson, and C. Morris. A survey on graph kernels. *Applied Network Science*, 5(1):1–42, 2020. Cited on page 3.

F. J. MacWilliams and N. J. A. Sloane. *The Theory of Error Correcting Codes*. Elsevier, 1977. Cited on page 5.

D. L. Mobley and J. P. Guthrie. FreeSolv: a database of experimental and calculated hydration free energies, with input files. *Journal of Computer-Aided Molecular Design*, 28(7):711–720, 2014. Cited on page 8.

G. Nikolentzos, G. Siglidis, and M. Vazirgiannis. Graph kernels: A survey. *Journal of Artificial Intelligence Research*, 72:943–1027, 2021. Cited on page 3.

R. O'Donnell. *Analysis of Boolean Functions*. Cambridge University Press, 2014. Cited on pages 4, 12.

S. Ovchinnikov. *Graphs and Cubes*. Springer Science & Business Media, 2011. Cited on page 11.

C. E. Rasmussen and C. K. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006. Cited on page 2.

D. J. Robinson. *An Introduction to Abstract Algebra*. de Gruyter, 2003. Cited on pages 3, 11.

N. Shervashidze, P. Schweitzer, E. J. Van Leeuwen, K. Mehlhorn, and K. M. Borgwardt. Weisfeiler-Lehman Graph Kernels. *Journal of Machine Learning Research*, 12(9), 2011. Cited on page 3.

J. Snoek, H. Larochelle, and R. P. Adams. Practical Bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems*, 2012. Cited on page 1.

N. Srinivas, A. Krause, S. M. Kakade, and M. W. Seeger. Gaussian Process Optimization in the Bandit Setting: No Regret and Experimental Design. In *International Conference on Machine Learning*, 2010. Cited on page 8.

J. Wilson, V. Borovitskiy, A. Terenin, P. Mostowsky, and M. Deisenroth. Efficiently sampling functions from Gaussian process posteriors. In *International Conference on Machine Learning*, 2020. Cited on page 2.

J. T. Wilson, V. Borovitskiy, A. Terenin, P. Mostowsky, and M. P. Deisenroth. Pathwise Conditioning of Gaussian Processes. *Journal of Machine Learning Research*, 22(105):1–47, 2021. Cited on page 2.

Z. Wu, B. Ramsundar, E. N. Feinberg, J. Gomes, C. Geniesse, A. S. Pappu, K. Leswing, and V. Pande. MoleculeNet: a benchmark for molecular machine learning. *Chemical Science*, 9(2):513–530, 2018. Cited on page 8.

A Gaussian Processes on Finite Spaces of Graphs

As in Section 2, define $\mathcal{V}$ to be one of the sets $\mathcal{U}_n, \mathcal{UL}_n, \mathcal{D}_n$, or $\mathcal{DL}_n$ of unweighted graphs on n nodes, identified with $\{0,1\}^d$ for the appropriate d. The metagraph $\mathcal{G}$ from Section 2 may be recognized to be the *hypercube graph* (Kondor and Lafferty, 2002), also referred to as the *d-cube graph* or just a *cube graph* (Ovchinnikov, 2011). We start by describing the group $\mathrm{Aut}(\mathcal{G})$ of its automorphisms. To do this, we need to introduce several notions from the group theory.

First, recall that a subgroup N of a group G is called *normal* if and only if $gxg^{-1} \in N$ for all $g \in G$ and $x \in N$. Now we formally introduce the semidirect product of groups (Robinson, 2003).

Definition. Given a group G, a subgroup $H \subseteq G$ and a normal subgroup $N \subseteq G$, we say that G is the *semidirect product of N and H* if and only if $G = NH$ and $N \cap H = e$ where $e \in G$ is the identity. In this case we write $G = N \rtimes H$.

Note that for all $g \in G = N \rtimes H$ there are unique $n \in N$ and $h \in H$ such that $g = nh$ (Robinson, 2003). With this we can characterize the group $\mathrm{Aut}(\mathcal{G})$ of automorphisms of the metagraph $\mathcal{G}$.

Result 1. *$\mathrm{Aut}(\mathcal{G}) = \mathbb{Z}_2^d \rtimes \mathrm{S}_d$ where the groups $\mathbb{Z}_2^d$ and S_d were introduced Section 2. This semidirect product has its own name, called the* hyperoctahedral group.[7]

Proof. See Ovchinnikov (2011, Theorem 3.31). □

This mirrors the Euclidean space, where the group of isometries of $\mathbb{R}^d$ is $\mathrm{E}(d) = \mathbb{R}^d \rtimes \mathrm{O}(d)$ where $\mathbb{R}^d$ is the Euclidean addition group acting on $\mathbb{R}^d$ by translations and $\mathrm{O}(d)$ is the group of orthogonal matrices acting on $\mathbb{R}^d$ by rotations. There is an even deeper connection here. The set $\{0,1\}$ endowed with addition and multiplication modulo 2 becomes the *field of characteristic* 2 (Dummit and Foote, 2004). The addition group of the corresponding vector space $\{0,1\}^d$ coincides with the group $\mathbb{Z}_2^d$, while all the orthogonal matrices over this field consist of only 0 or 1 entries, meaning that they are permutation matrices, turning the group $\mathrm{O}(d)$ into S_d in this case.

We are now ready to prove Theorem 1 from Section 2.

Theorem 1. *A Gaussian process $f \sim \mathrm{GP}(0, k)$ on $\mathcal{V}$ is isotropic if and only if for all $x, y, z \in \mathcal{V} = \mathbb{Z}_2^d$ and $\sigma \in \mathrm{S}_d$*

$$\begin{aligned} (i): &\qquad k(x \dot{+} z, y \dot{+} z) = k(x, y), \\ (ii): &\qquad k(\sigma(x), \sigma(y)) = k(x, y). \end{aligned}$$

Proof. This is an almost immediate corollary of Result 1. Since all translations $\cdot \to \cdot \dot{+} z$ and all permutations $\cdot \to \sigma(\cdot)$ are automorphisms of the metagraph $\mathcal{G}$, the forward implication is obvious.

The backward implication follows from the fact that each element $g \in \mathrm{Aut}(\mathcal{G})$ can be uniquely represented as a product th where t is a translation by some $z \in \mathbb{Z}_2^d$ and h is a permutation $\sigma \in \mathrm{S}_n$, this is a basic property of the semidirect product $\mathbb{Z}_2^d \rtimes \mathrm{S}_d$. □

We will further study the metagraph $\mathcal{G}$ through the following notion.

Definition. Let G be a group, and let $S \subseteq G$ be such subset that $S = S^{-1}$ and $e \notin S$, where $e \in G$ is the identity element. Then the *Cayley graph* $\mathrm{Cayley}(G, S)$ is the graph with vertex set $V = G$ and with the edge set E such that $(g, g') \in E$ if and only if $g'g^{-1} \in S$ (Godsil and Royle, 2001).

Define $\boldsymbol{e}_i \in \mathbb{Z}_2^d$ to be the vectors of zeroes and ones where there is a single 1, standing at the ith position. Denote $S = \{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_d\}$. Our metagraph $\mathcal{G}$, the hypercube graph, may be recognized to be the Cayley graph $\mathrm{Cayley}(\mathbb{Z}_2^d, S)$ of the group $\mathbb{Z}_2^d$ with this specific S (Godsil and Royle, 2001).

Consider the space $L^2(\mathbb{Z}_2^d)$ of real-valued functions $f : \mathbb{Z}_2^d \to \mathbb{R}$ with inner product given by

$$\langle f, g \rangle_{L^2(\mathbb{Z}_2^d)} = \frac{1}{2^d} \sum_{x \in \mathbb{Z}_2^d} f(x) f(x) \tag{31}$$

[7] The hyperoctahedral group is defined as the *wreath product* $\mathrm{S}_2 \wr \mathrm{S}_d = \mathrm{S}_2^d \rtimes \mathrm{S}_d = \mathbb{Z}_2^d \rtimes \mathrm{S}_d$ (Kerber, 2006).

This space admits (O'Donnell, 2014) an orthonormal basis of *characters* of the group $\mathbb{Z}_2^d$, i.e., functions $\chi : \mathbb{Z}_2^d \to \{0, 1\}$ with $\chi(x \dot{+} y) = \chi(x)\chi(y)$. Moreover, this basis is known to be the family $\{w_T\}_{T\subseteq\{1,\dots,d\}}$ of Walsh functions given by

$$w_T(x) = (-1)^{\sum_{t\in T} x_t} \in \{-1, 1\} \tag{32}$$

Note that the number of such functions is 2^d that matches the dimension of $L^2(\mathbb{Z}_2^d)$. Representing functions from $L^2(\mathbb{Z}_2^d)$ in terms of this basis is similar to representing periodic functions in form of the classical Fourier series.

Consider the adjacency matrix $\mathbf{A}$ of $\mathcal{G} = \mathrm{Cayley}(\mathbb{Z}_2^d, S)$. We may interpret it as the operator

$$\mathbf{A} : L^2(\mathbb{Z}_2^d) \to L^2(\mathbb{Z}_2^d) \qquad \text{given by} \qquad (\mathbf{A}f)(x) = \sum_{y\in\mathbb{Z}_2^d} \mathbf{A}_{xy} f(y). \tag{33}$$

Now we prove the following auxiliary result.

Lemma 1. *The Walsh functions $w_T : \mathbb{Z}_2^d \to \mathbb{R}$ are eigenfunctions of the operator $\mathbf{A}$ corresponding to eigenvalues $\lambda_T = d - 2|T|$. That is $\mathbf{A}w_T = \lambda_T w_T$ with $\lambda_T = d - 2|T|$.*

Proof. For $x \in \mathbb{Z}_2^d$ its inverse $\dot{-}\, x$ is equal to x itself, hence the condition $g'g^{-1} \in S$ from the definition of a Cayley graph turns into $y \dot{-} x = y \dot{+} x = \boldsymbol{e}_i$ for some $i \in \{1, \dots, d\}$. We have

$$(\mathbf{A}w_T)(x) = \sum_{y\in\mathbb{Z}_2^d} \mathbf{A}_{xy} w_T(y) = \sum_{i=1}^{d} w_T(x \dot{+} \boldsymbol{e}_i) = w_T(x) \sum_{i=1}^{d} w_T(\boldsymbol{e}_i). \tag{34}$$

Write

$$w_T(\boldsymbol{e}_i) = (-1)^{\sum_{t\in T}(\boldsymbol{e}_i)_t} = \begin{cases} -1 & i \in T \\ 1 & \text{otherwise.} \end{cases} \tag{35}$$

Hence, $\lambda_T = \sum_{i=1}^{d} w_T(\boldsymbol{e}_i) = (d - |T|) - |T| = d - 2|T|$ which proves the claim. □

Now we are ready to prove Theorem 3

Theorem 3. *The set $\{w_T(\cdot)\}_{T\subseteq\{1,\dots,d\}}$ is an orthonormal basis of $L^2(\mathcal{V})$ consisting of eigenfunctions of any of the Laplacians $\boldsymbol{\Delta}, \boldsymbol{\Delta}_{rw}, \boldsymbol{\Delta}_{sym}$ on $\mathcal{G}$ with eigenvalues*

$$\lambda_T^{\boldsymbol{\Delta}} = 2|T| \qquad\qquad \lambda_T^{\boldsymbol{\Delta}_{rw}} = \lambda_T^{\boldsymbol{\Delta}_{sym}} = 2|T|/d \tag{10}$$

Proof. Recall that the Laplacian matrix $\boldsymbol{\Delta}$ is given by $\boldsymbol{\Delta} = \mathbf{D} - \mathbf{A}$ where $\mathbf{D}$ is the diagonal degree matrix. For the metagraph $\mathcal{G}$ we have $\mathbf{D} = d\mathbf{I}$. It follows that $\boldsymbol{\Delta}$, interpreted as the operator of the form $\boldsymbol{\Delta} : L^2(\mathbb{Z}_2^d) \to L^2(\mathbb{Z}_2^d)$ has $\{w_T\}_{T\subseteq\{1,\dots,d\}}$ as its basis of eigenfunctions and

$$\lambda_j = d - d + 2|T| = 2|T| \tag{36}$$

as the corresponding eigenvalues.

Recall that $\boldsymbol{\Delta}_{rw} = \mathbf{I} - \mathbf{D}^{-1}\mathbf{A}$ and $\boldsymbol{\Delta}_{sym} = \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$ which in our case means that $\boldsymbol{\Delta}_{rw} = \boldsymbol{\Delta}_{sym} = \boldsymbol{\Delta}/d$, hence the remaining part of the claim. □

With these considerations, we are ready to prove the remaining theorems of Section 2.

Theorem 2. *$f \sim \mathrm{GP}(0, k)$ is isotropic on $\mathcal{V}$ if and only if*

$$k(x, y) = \sum_{j=0}^{d} \alpha_j \sum_{T\subseteq\{1,\dots,d\}, |T|=j} w_T(x) w_T(y), \qquad\qquad \alpha_j \geq 0, \tag{8}$$

where $w_T(x) = (-1)^{\sum_{t\in T} x_t}$ are the Walsh functions — the analogs of complex exponentials in the Fourier analysis of Boolean functions (O'Donnell, 2014).

Proof. Since $\{w_T\}_{T\subseteq\{1,\dots,d\}}$ is an orthonormal basis of $L^2(\mathbb{Z}_2^d)$ we can write

$$f(x) = \sum_{T\subseteq\{1,\dots,d\}} \varepsilon_T w_T(x), \qquad \varepsilon_T = \langle f, w_T\rangle = \frac{1}{2^d}\sum_{y\in Z_2^d} f(y)w_T(y), \tag{37}$$

where ε_T are Gaussian random variables. Of course $\mathbb{E}\,\varepsilon_T = 0$. Write

$$\operatorname{Cov}(\varepsilon_T, \varepsilon_{T'}) = \frac{1}{2^{2d}}\sum_{x,y\in Z_2^d} \operatorname{Cov}(f(x), f(y))w_T(x)w_{T'}(y) \tag{38}$$

$$= \frac{1}{2^{2d}}\sum_{x,y\in Z_2^d} k(x,y)w_T(x)w_{T'}(y) \tag{39}$$

Note that for any $g \in L^2(Z_2^d)$ and any $z \in \mathbb{Z}_2^d$ we have $\sum_{y\in\mathbb{Z}_2^d} g(z \dot{+} y) = \sum_{y\in\mathbb{Z}_2^d} g(y)$. Denoting $\mathbf{0}$ to be the graph with no edges, we have

$$\operatorname{Cov}(\varepsilon_T, \varepsilon_{T'}) = \frac{1}{2^{2d}}\sum_{x\in Z_2^d}\sum_{y\in Z_2^d} k(x, x\dot{+}y)w_T(x)w_{T'}(x\dot{+}y) \tag{40}$$

$$= \frac{1}{2^{2d}}\sum_{x\in Z_2^d}\sum_{y\in Z_2^d} k(\mathbf{0}, y)w_T(x)w_{T'}(x)w_{T'}(y) \tag{41}$$

$$= \Big(\frac{1}{2^d}\sum_{x\in Z_2^d} w_T(x)w_{T'}(x)\Big)\Big(\frac{1}{2^d}\sum_{y\in Z_2^d} k(\mathbf{0}, y)w_{T'}(y)\Big). \tag{42}$$

Since w_T are orthonormal, the last equation shows that $\operatorname{Cov}(\varepsilon_T, \varepsilon_{T'}) = 0$ for $T \neq T'$. Hence all ε_T are independent. If we denote their respective variances $\operatorname{Var}(\varepsilon_T)$ by α_T, we get

$$k(x,y) = \sum_{T\subseteq\{1,\dots,d\}} (\operatorname{Var}(\varepsilon_T))w_T(x)w_T(y) = \sum_{T\subseteq\{1,\dots,d\}} \alpha_T w_T(x)w_T(y). \tag{43}$$

Now recall that $w_T(x) = (-1)^{\sum_{t\in T} x_t}$. For $\sigma \in \mathrm{S}_d$ write

$$w_T(\sigma(x)) = (-1)^{\sum_{t\in T} x_{\sigma(t)}} = (-1)^{\sum_{t\in\sigma(T)} x_t} = w_{\sigma(T)}(x) \tag{44}$$

where $\sigma(T)$ denotes the action of σ on the set T elementwise. It follows that

$$k(\sigma(x), \sigma(y)) = \sum_{T\subseteq\{1,..,d\}} \alpha_T w_{\sigma(T)}(x)w_{\sigma(T)}(y) \tag{45}$$

Since $\{w_T\}_{T\subseteq\{1,\dots,d\}}$ is an orthonormal basis and since $k(x,y) = k(\sigma(x),\sigma(y))$ by assumption, we have $\alpha_T = \alpha_{\sigma(T)}$ for all $\sigma \in \mathrm{S}_d$. Hence, $\alpha_T = \alpha_{T'}$ for all $T, T' \subseteq \{1,\dots,d\}$ such that $|T| = |T'|$. This proves the claim. □

Recall that $|x| = \sum_{j=1}^d x_d$ and

$$G_{d,j}(x,y) = \sum_{T\subseteq\{1,\dots,d\}, |T|=j} w_T(x)w_T(y). \tag{46}$$

We now prove the recurrence relation governing the values of $G_{d,j}$, enabling us to use a dynamical program to find them.

Theorem 4. *$G_{d,j}(x,y) = G_{d,j,m}$ where $m = \left|x \dot{+} y\right|$ is the Hamming distance between graphs x and y. Moreover,*

$$G_{d,j,m} = G_{d-1,j,m-1} - G_{d-1,j-1,m-1} \tag{12}$$

with $G_{1,j,m} = (-1)^m$, $G_{d,0,m} = 1$ and $G_{d,j,0} = \binom{d}{j}$.

Proof. First of all, by direct computatiom, $G_{d,0,m} = 1$, $G_{1,j,m} = (-1)^m$.

Denote $z = x \dot{+} y$ and write

$$G_{d,j}(x,y) = \sum_{T\subseteq\{1,\dots,d\},|T|=j} w_T(x)w_T(y) = \sum_{T\subseteq\{1,\dots,d\},|T|=j} w_T(x \dot{+} y) = \sum_{T\subseteq\{1,\dots,d\},|T|=j} w_T(z). \tag{47}$$

Suppose z has m ones and $d-m$ zeros. Since the kernel is invariant with respect to all edge permutations, we can assume that $z = (1,\dots,1,0,\dots,0)$, i.e., z has m ones followed by $d-m$ zeros.

The terms in the sum above are $(-1)^\ell$ if $|T \cap \{1,\dots,m\}| = \ell$. Hence,

$$G_{d,j}(x,y) = G_{d,j,m} = \sum_{\ell=\max(0,m+j-d)}^{\min(j,m)} (-1)^\ell \binom{m}{\ell}\binom{d-m}{j-\ell}. \tag{48}$$

Specifically, if $m = 0$, then $G_{d,j,0} = \binom{d}{j}$.

One can also construct a recurrence relation as follows. Recall that we assume that z has $m > 0$ ones followed by $d-m$ zeros. Dividing subsets T of size j into two classes, those that include index 1 and those that do not, we write

$$\begin{aligned} G_{d,j}(x,y) &= \sum_{T:|T|=j,1\in T} w_T(z) + \sum_{T:|T|=j,1\notin T} w_T(z) \\ &= (-1)G_{d-1,j-1,m-1} + G_{d-1,j,m-1}. \end{aligned}$$

Thus, $G_{d,j,m} = G_{d-1,j,m-1} - G_{d-1,j-1,m-1}$. □

Note that $G_{d,j,m}$ may become quite large, this is already apparent from $G_{d,j,0} = \binom{d}{j}$. Because of this, when implementing the dynamical program, it makes sense to consider $G'_{d,j,m} = G_{d,j,m}/\binom{d}{j}$ for which we have $G'_{d,0,m} = 1$, $G'_{1,j,m} = (-1)^m$ same as before, but now $G'_{d,j,0} = 1$ and

$$G'_{d,j,m} = G_{d,j,m}/\binom{d}{j} = G_{d-1,j,m-1}/\binom{d}{j} - G_{d-1,j-1,m-1}/\binom{d}{j} \tag{49}$$

$$= \frac{d-j}{d} G_{d-1,j,m-1}/\binom{d-1}{j} - \frac{j}{d} G_{d-1,j-1,m-1}/\binom{d-1}{j-1} \tag{50}$$

$$= \frac{d-j}{d} G'_{d-1,j,m-1} - \frac{j}{d} G'_{d-1,j-1,m-1}. \tag{51}$$

B Gaussian Processes on Spaces of Graph Equivalence Classes

Recall that $\mathcal{V}$ denotes one of the sets $\mathcal{U}_n, \mathcal{UL}_n, \mathcal{D}_n, \mathcal{DL}_n$ of unweighted graphs on n nodes, identified with $\{0,1\}^d$ for the appropriate d. Recall that $H \subseteq \mathrm{S}_n$ denotes a subgroup of the node permutation group S_n. This H induces the equivalence relation $\sim$ and the set of $\sim$-equivalence classes $\mathcal{V}_{/H}$. Recall that $\overline{x} \in \mathcal{V}_{/H}$ denotes the equivalence class of the element $x \in \mathcal{V}$.

Virtually all the properties of H-invariant versions of Gaussian processes and their kernels will be the consequences of the fact that the partition $\mathcal{V} = \cup_{\overline{x}\in\mathcal{V}_{/H}} \overline{x}$ is *equitable*. We prove this, after formally introducing this notion and presenting its relevant properties, in the following subsection.

B.1 Equitable Partitions

Consider an unweighted undirected graph $G = (V,E)$ with adjacency matrix $\mathbf{A}_G$.

Definition. For a set $C \subseteq V$ and a vertex $v \in V$ define $\deg(v,C) = |\{v' \in C : (v,v') \in E\}|$. A partition $V = \cup_{i=1}^m V_i$ is called *equitable* if and only if

$$\deg(v,V_i) = \deg(v',V_i) \qquad \text{for all } v,v' \in V_j \qquad \text{and all } j \in \{1,\dots,m\}. \tag{52}$$

We may consider the adjacency matrix of G as an operator $\mathbf{A}_G : L^2(V) \to L^2(V)$.

Theorem 7. *Consider an equitable partition $V = \cup_{i=1}^m V_i$. There is an orthonormal basis $\{f_j\}_{j=1}^{|V|}$ of $L^2(V)$ consisting of eigenfunctions of $\mathbf{A}_G$ split in two groups: $\{1, \ldots, |V|\} = \Lambda_1 \cup \Lambda_2$ such that*

$$f_j\big|_{V_i} \equiv c_{ij} \quad \text{if} \quad j \in \Lambda_1, \qquad \sum_{v \in V_i} f_j(v) = 0 \quad \text{if} \quad j \in \Lambda_2, \qquad 1 \leq i \leq m. \tag{53}$$

Moreover, $|\Lambda_1| = m$ and $|\Lambda_2| = |V| - m$. In words, there are m functions in Λ_1, all of which are piecewise-constant on the partition, and any function in Λ_2 has zero average in each part of the partition.

Proof. This is a widely known fact (see e.g., Cozzo et al. (2018)), we present here its simple proof, as it is instructive.

Define $\mathbb{1}_{V_i} : V \to \{0, 1\}$ to be the indicator of the set V_i, i.e., $\mathbb{1}_{V_i}(v) = 1$ if $v \in V_i$ and $\mathbb{1}_{V_i}(v) = 0$ otherwise. Let us denote by $W = \operatorname{span}\{\mathbb{1}_{V_i}\}_{i=1}^m \subseteq L^2(V)$ the space of functions constant on all sets V_j. It is easy to check that

$$(\mathbf{A}_G \mathbb{1}_{V_i})(v) = \deg(v, V_i). \tag{54}$$

Since the partition is equitable, the right-hand side, as a function of v, is constant on all sets V_j and thus $\mathbf{A}_G \mathbb{1}_{V_i} \in W$. This means that W is an invariant subspace of the operator $\mathbf{A}_G$.

Since $\mathbf{A}_G$, as a matrix, is symmetric, there exists an orthonormal basis of $L^2(V)$ consisting of its eigenvectors (eigenfunctions). From classical linear algebra we know that it may be chosen to consist of vectors split in two groups: vectors belonging to the invariant space W and vectors belonging to its orthogonal complement $W^\perp$. The former are constant on all sets V_j, meaning that for them the left part of Equation (53) holds, while the latter, since orthogonal to W, average to zero over all sets V_j, i.e., the right part of Equation (53) holds for them. Of course $\dim W = m$ and $\dim W + \dim W^\perp = |V|$. This proves the claim. □

Now we turn to the metagraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ from Section 2 and prove that the partition of its vertex set $\mathcal{V}$ into $\sim_H$-equivalence classes is equitable.

Proposition 1. *For any subgroup $H \subseteq \mathrm{S}_n$ the partition $\mathcal{V} = \cup_{\overline{x} \in \mathcal{V}_{/H}} \overline{x}$ generated by the equivalence relation $\sim_H$ is equitable.*

Proof. If $x, x' \in \overline{x}$ then $x' = \sigma(x)$ for some permutation $\sigma \in H$. If $(x, y) \in \mathcal{E}$ then x and y differ by a single edge, say ith one. Obviously, then $\sigma(x)$ and $\sigma(y)$ differ by a single edge, the $\sigma(i)$th one. Thus, of course $(x', \sigma(y)) \in \mathcal{E}$. The converse also holds: if $(x', \sigma(y))$ is an edge, then (x, y) is an edge because $\sigma^{-1} \in H$. It follows that $\deg(x, \overline{y}) = \deg(x', \overline{y})$ for all $\overline{y} \in \mathcal{V}_{/H}$. □

With this, we have necessary tools to prove Theorems 5 and 6 of Section 3, starting with the former.

B.2 Φ-kernels on Quotient Graphs and the Proof of Theorem 5

Put $h = |\mathcal{V}_{/H}|$. Let us enumerate graphs in $\mathcal{V}$ by numbers from $1 \leq i \leq 2^d$ and equivalence classes in $\mathcal{V}_{/H}$ by numbers from $1 \leq j \leq h$: denote the ith graph by $g(i)$ and the jth class by $c(j)$.

We start by proving the following elementary lemma.

Lemma 2. *Take $x, z \in \mathcal{V}$ such that $z \in \overline{x}$. Define*

$$\alpha(x, z) = |\{\sigma \in H : \sigma(x) = z\}| \tag{55}$$

Then $\alpha(x, z) = \alpha(x', z')$ for all $x', z' \in \overline{x}$. In particular, $\alpha(x, z) = \alpha(x, x)$.

Proof. We have

$$\alpha(z, x) = |\{\sigma \in H : \sigma(z) = x\}| = \left|\left\{\sigma \in H : \sigma^{-1}(x) = z\right\}\right| = \alpha(x, z) \tag{56}$$

hence it is enough to prove that $\alpha(x, z) = \alpha(x, x)$. Since $z \in \overline{x}$, we have $z = \sigma_z(x)$. Write

$$\begin{aligned} \alpha(x, z) &= |\{\sigma \in H : \sigma(x) = \sigma_z(x)\}| = \left|\left\{\sigma \in H : \sigma_z^{-1}(\sigma(x)) = x\right\}\right| && (57) \\ &= \left|\left\{\sigma_z^{-1}\sigma \in \sigma_z^{-1}H : \sigma_z^{-1}(\sigma(x)) = x\right\}\right| && (58) \\ &= \left|\left\{\sigma_z^{-1}\sigma \in H : \sigma_z^{-1}(\sigma(x)) = x\right\}\right| = |\{\sigma \in H : \sigma(x) = x\}| = \alpha(x, x). \end{aligned}$$

□

With this, we are ready to prove that $\Pr$ from Section 3 is indeed an orthonormal projector. First, it is obvious that $\Pr \Pr f = \Pr f$ for all $f \in L^2(\mathcal{V})$ and thus $\Pr$ is a projection. Recall that W denotes the subspace of $L^2(\mathcal{V})$ consisting of functions that are constant on all equivalence classes. It is easy to see that $\Pr f \in W$ for all $f \in L^2(\mathcal{V})$ and $\Pr f = f$ if $f \in W$. Finally, let us prove that if $f \in W^\perp$, i.e., if $\sum_{x \in c(j)} f(x) = 0$ for all $j = 1, \ldots, h$, then $\Pr f = 0$. Write

$$(\Pr f)(x) = \frac{1}{|H|} \sum_{\sigma \in H} f(\sigma(x)) = \frac{1}{|H|} \sum_{z \in \overline{x}} f(z) \alpha(x, z) \tag{59}$$

$$= \frac{1}{|H|} \sum_{z \in \overline{x}} f(z) \alpha(x, x) = \frac{\alpha(x, x)}{|H|} \sum_{z \in \overline{x}} f(z) = 0. \tag{60}$$

We will use this to verify Theorem 5 of Section 3.

Theorem 5. *Consider the class of Φ-kernels induced by the symmetric normalized Laplacian, and take a Gaussian process $f \sim \mathrm{GP}(0, k)$ on $\mathcal{V}$ with a Φ-kernel given by a function Φ. Its H-invariant version $f_{/H} \sim \mathrm{GP}(0, k_{/H})$ has kernel $k_{/H}(x, y) = \psi(\overline{x})\psi(\overline{y}) k_\Phi(\overline{x}, \overline{y})$ where k_Φ is the Φ-kernel on the quotient graph $\mathcal{G}_{/H}$ with the same function Φ. Moreover, $\psi(\overline{x}) = |\overline{x}|^{-1/2}$.*

Proof. Consider the adjacency matrix $\mathbf{A}$ of $\mathcal{G}$ as an operator $\mathbf{A} : L^2(\mathcal{V}) \to L^2(\mathcal{V})$ and use Theorem 7 to choose an orthonormal basis $\{f_j\}_{j=1}^{2^d}$ of eigenfunctions of $\mathbf{A}$ such that $f_1, \ldots, f_h$ are constant on all equivalence classes and the rest average to zero over all equivalence classes. Denote the eigenvalues of $\mathbf{A}$ corresponding to f_j by λ_j and the eigenvalues of the symmetric normalized Laplacian $\boldsymbol{\Delta}_{sym}$ by $\lambda_j^{sym} = 1 - \lambda_j / d$. Then a Φ-class kernel k on $\mathcal{G}$ is given by

$$k(x, y) = \sum_{j=1}^{h} \Phi(\lambda_j^{sym}) f_j(x) f_j(y) + \sum_{j=h+1}^{2^d} \Phi(\lambda_j^{sym}) f_j(x) f_j(y). \tag{61}$$

Because $\Pr$ is an orthonormal projector onto the space W, we have $\Pr f_j = f_j$ for $j \in \{1, \ldots, h\}$ and $\Pr f_j = 0$ for $j > h$. With this, we can write

$$k_{/H}(x, y) = \sum_{j=1}^{h} \Phi(\lambda_j^{sym}) (\Pr f_j)(x) (\Pr f_j)(y) \tag{62}$$

$$+ \sum_{j=h+1}^{2^d} \Phi(\lambda_j^{sym}) (\Pr f_j)(x) (\Pr f_j)(y) \tag{63}$$

$$= \sum_{j=1}^{h} \Phi(\lambda_j^{sym}) f_j(x) f_j(y). \tag{64}$$

Let us denote the adjacency matrix of the weighted graph $\mathcal{G}_{/H} = (\mathcal{V}_{/H}, \mathcal{E}_{/H})$ by $\mathbf{A}_{/H}$. Define also the $2^d \times h$ matrix $\mathbf{S}$ with $\mathbf{S}_{ij} = 1$ if $g(i) \in c(j)$ and $\mathbf{S}_{ij} = 0$ otherwise. Then we have

$$(\mathbf{A}_{/H})_{ij} = \sum_{x \in c(i)} \sum_{y \in c(j)} \mathbf{A}_{xy} \qquad \text{and thus} \qquad \mathbf{A}_{/H} = \mathbf{S}^\top \mathbf{A} \mathbf{S}. \tag{65}$$

Computing the corresponding degree matrix yields

$$\big(\mathbf{D}_{/H}\big)_{ii} = \sum_{j=1}^{h} \big(\mathbf{A}_{/H}\big)_{ij} = \sum_{j=1}^{h} \sum_{x \in c(i)} \sum_{y \in c(j)} \mathbf{A}_{xy} = \sum_{x \in c(i)} \sum_{y \in \mathcal{V}} \mathbf{A}_{xy} = |c(i)| d. \tag{66}$$

If we introduce the diagonal matrix $\mathbf{N}$ with $\mathbf{N}_{ii} = |c(i)|$, then $\mathbf{D}_{/H} = d\mathbf{N}$. Note that we have $\mathbf{S}\mathbf{N}^{-1}\mathbf{S}^\top f_j = f_j$. Define

$g_j = \mathbf{N}^{-1/2}\mathbf{S}^\top f_j \in L^2(\mathcal{V}_{/H})$ and $\mathbf{\Delta}_{sym/H} = \mathbf{I} - \mathbf{D}_{/H}^{-1/2}\mathbf{A}_{/H}\mathbf{D}_{/H}^{-1/2}$. We thus have

$$\begin{aligned}
\mathbf{\Delta}_{sym/H}\, g_j &= g_j - \mathbf{D}_{/H}^{-1/2}\mathbf{S}^\top \mathbf{A}\mathbf{S}\mathbf{D}_{/H}^{-1/2}\mathbf{N}^{-1/2}\mathbf{S}^\top f_j && (67)\\
&= g_j - \mathbf{D}_{/H}^{-1/2}\mathbf{S}^\top \mathbf{A} d^{-1/2}\mathbf{S}\mathbf{N}^{-1}\mathbf{S}^\top f_j && (68)\\
&= g_j - \mathbf{D}_{/H}^{-1/2}\mathbf{S}^\top \mathbf{A} d^{-1/2} f_j && (69)\\
&= g_j - \mathbf{D}_{/H}^{-1/2}\mathbf{S}^\top \lambda_j d^{-1/2} f_j && (70)\\
&= g_j - d^{-1}\lambda_j \mathbf{N}^{-1/2}\mathbf{S}^\top f_j && (71)\\
&= g_j - d^{-1}\lambda_j g_j = (1 - d^{-1}\lambda_j) g_j = \lambda_j^{sym} g_j. && (72)
\end{aligned}$$

Hence, the Φ-kernel k_Φ on $\mathcal{G}_{/H}$ corresponding to the same Φ and to the symmetric normalized Laplacian is given by

$$k_\Phi(\overline{x}, \overline{y}) = \sum_{j=1}^{h} \Phi(\lambda_j^{sym}) g_j(\overline{x}) g_j(\overline{y}). \tag{73}$$

Denote $\psi(\overline{x}) = |\overline{x}|^{-1/2}$. It is easy to see by definition of g_j that we have $f_j(x) = \psi(\overline{x}) g_j(\overline{x})$ for indices $j \in \{1, \ldots, h\}$. Thus, obviously,

$$k_{/H}(x, y) = \psi(\overline{x})\psi(\overline{y}) k_\Phi(\overline{x}, \overline{y}) \tag{74}$$

which proves the claim. □

B.3 The Hardness Result

In this section we prove Theorem 6. We start with a simple lemma inspired by Gärtner et al. (2003).

Lemma 3. *Consider a kernel $k : \mathcal{V} \times \mathcal{V} \to \mathbb{R}$ such that $k(\sigma_1(x), \sigma_2(y)) = k(x, y)$ for all permutations $\sigma_1, \sigma_2 \in H$. Define the kernel $\tilde{k} : \mathcal{V}_{/H} \times \mathcal{V}_{/H} \to \mathbb{R}$ by $\tilde{k}(\overline{x}, \overline{y}) = k(x, y)$ and assume that $\tilde{k}(\overline{x}, \overline{y}) = \langle \phi(\overline{x}), \phi(\overline{y}) \rangle$ for a certain feature map $\phi : \mathcal{V}_{/H} \to \mathbb{R}^l$, $l \in \mathbb{N}$. If ϕ is injective, i.e., if $\phi(\overline{x}) = \phi(\overline{y})$ implies $\overline{x} = \overline{y}$, then $k(x, y) = (k(x, x) + k(y, y))/2$ implies $x \sim y$.*

Proof. Write

$$\begin{aligned}
k(x, x) + k(y, y) - 2k(x, y) &= \tilde{k}(\overline{x}, \overline{x}) + \tilde{k}(\overline{y}, \overline{y}) - 2\tilde{k}(\overline{x}, \overline{y}) && (75)\\
&= \langle \phi(x) - \phi(y), \phi(x) - \phi(y) \rangle && (76)\\
&= \|\phi(x) - \phi(y)\|^2 && (77)
\end{aligned}$$

This means that $k(x, y) = (k(x, x) + k(y, y))/2$ is equivalent to $\phi(\overline{x}) = \phi(\overline{y})$, which is by assumption is equivalent to $\overline{x} = \overline{y}$, i.e., $x \sim y$. □

It is therefore enough to evaluate such a kernel k at three pairs of inputs to check whether $x \sim y$. In particular, if $\sim$ is the graph isomorphism relation $\cong$ (i.e., $H = \mathrm{S}_n$), this shows that computing the kernel pointwise is at least as hard as resolving the graph isomorphism problem.

We prove Theorem 6 by showing that the H-invariant version of a Φ-kernel corresponding to a strictly positive Φ may be represented via an injective feature map, a consequence of the partition $\mathcal{V} = \cup_{\overline{x} \in \mathcal{V}_{/H}} \overline{x}$ of the set of graphs into sets of equivalence classes being equitable.

Theorem 6. *Assume k is a Φ-kernel on $\mathcal{G}$ for a strictly positive Φ. Exactly computing $k_{/H}$ at three pairs of inputs $(x, y), (x, x)$ and (y, y) is at least as hard as checking whether $x \sim_H y$. In particular, if $H = \mathrm{S}_n$, it is at least as hard as resolving the graph isomorphism problem.*[8]

[8] Whether or not this may be done in polynomial time is currently considered an open problem.

Proof. From Theorem 5 we know that $k_{/H} = \psi(x)\psi(y)k_\Phi(\overline{x}, \overline{y})$ where k_Φ is the Φ-kernel on the quotient graph $\mathcal{G}_{/H}$ corresponding to the the symmetric normalized Laplacian and the same Φ as k. Recall the notation $h = \left|\mathcal{V}_{/H}\right|$. It follows that for an orthonormal basis $\{f_j\}_{j=1}^h \in L^2(\mathcal{V}_{/H})$ of eigenfunctions $\boldsymbol{\Delta}_{sym} f_j = \lambda_j f_j$ we have

$$k_{/H}(x, y) = \sum_{j=1}^{h} \Phi(\lambda_j)\psi(\overline{x})f_j(\overline{x})\psi(\overline{y})f_j(\overline{y}) \qquad \Phi(\lambda_j) > 0. \tag{78}$$

Consider the function $\tilde{k}_{/H} : \mathcal{V}_{/H} \times \mathcal{V}_{/H} \to \mathbb{R}$ given by $\tilde{k}_{/H}(\overline{x}, \overline{y}) = k_{/H}(x, y)$. Then, by Equation (78), it may be represented as $\tilde{k}_{/H}(\overline{x}, \overline{y}) = \langle \phi(\overline{x}), \phi(\overline{y}) \rangle$ with $\boldsymbol{\phi} : \mathcal{V}_{/H} \to \mathbb{R}^h$ given by

$$\boldsymbol{\phi}(\overline{x}) = \Big(\Phi(\lambda_1)^{1/2}\psi(\overline{x})f_1(\overline{x}), \ldots, \Phi(\lambda_h)^{1/2}\psi(\overline{x})f_h(\overline{x})\Big)^\top. \tag{79}$$

We now prove that $\boldsymbol{\phi}$ is injective. Since the functions f_j form an orthonormal basis in $L^2(\mathcal{V}_{/H})$, the functions $f'_j(\overline{x}) = \Phi(\lambda_j)^{1/2}\psi(\overline{x})f_j(\overline{x})$ form a linear basis of the same space. If we assume that $\boldsymbol{\phi}(\overline{x}) = \boldsymbol{\phi}(\overline{y})$ for $\overline{x} \neq \overline{y}$, then the function $\mathbb{1}_{\overline{x}} : \mathcal{V}_{/H} \to \mathbb{R}$ is not representable in the this basis because for all $f \in \operatorname{span}\{f'_j\}_{j=1}^h$ our assumption implies $f(\overline{x}) = f(\overline{y})$, hence a contradiction. Now the claim follows from Lemma 3 and the remarks afterwards by virtue of $\boldsymbol{\phi}$ being injective. □

C Additional Experimental Details

C.1 Computational Considerations

To ensure numerical stability during the dynamic program pre-computation (12), we use a normalizing factor of $\binom{d}{j}$ (as done in (49)). This factor must be accounted for in the kernel evaluation. We normalize the kernel such that $k(x, x) = 1$ to ensure that $\binom{d}{j}$ factor does not cause numerical instability during hyperparameter optimization. In code, the kernel normalization is performed in the log scale using the Log-Sum-Exp (LSE) trick.

While some of the computations, especially those over possible hamming distances in the PROJECTED case, can be made more efficient, the preliminary version of our implementation precomputes the hamming distances between possible permutations over equivalence classes in the PROJECTED and caches them to be reused during training and hyperparameter optimization.

C.2 Data Preparation

We use the FREESOLV dataset as provided by `Pytorch Geometric` under the `MoleculeNet` dataset. The dataset consists of 642 molecules with experimentally determined hydration free energy values. We remove invalid molecules, and molecules with only a single atom, and follow a random train/test split of 80/20 to obtain 510 examples for training and 128 examples for testing. The limited data setting here matches the typical application settings for Gaussian processes.

To test the exact projected Gaussian processes, we consider a smaller subset of FREESOLV (FREESOLV-S). Here, we allow molecules whose constituent atoms are within the allowed set of four atom types (carbon, nitrogen, oxygen, chlorine), with a maximum of 3 atoms per atom type. Following a similar train/test split of 80/20 gives us 52 training and 13 text examples.

For both FREESOLV & FREESOLV-S, the data normalized by subtracting the mean and dividing by the corresponding standard deviation. The normalization is only performed for the hydration free energies.

C.3 Hyperparameter Tuning

We optimize the hyperparameters of the kernel during training. For the Heat kernel in both the GRAPH and PROJECTED settings, the parameters κ and σ^2 are optimized. For the Matérn kernel under similar settings, we also optimized over the additional parameter ν. We experimented with initial values of 1.0 and 2.0 for κ for both the Heat and Matérn kernels, and observed that the training procedure converged to the same results for both situations. During model training, we also used learning rates of 0.1 and 0.001, with the higher learning rate improving the speed of convergence in most settings.

Another phenomenon we noticed was that the Matérn kernel for typical ν values of $0.5, 1.5$ or 2.5, decays very quickly, especially as n (and d) increase. The decay behavior is more reasonable for larger values of ν. In our experiments, we therefore set $\nu = \frac{d}{2} + \nu_{base}$ where $\nu_{base} \in \{0.5, 1.5, 2.5\}$. Figure 3 shows this decaying behavior of ν for different n.