Linearly Constrained Gaussian Processes with Boundary Conditions

Markus Lange-Hegermann
Department of Electrical Engineering and Computer Science
OWL University of Applied Sciences and Arts
markus.lange-hegermann@th-owl.de

Abstract

One goal in Bayesian machine learning is to encode prior knowledge into prior distributions, to model data efficiently. We consider prior knowledge from systems of linear partial differential equations together with their boundary conditions. We construct multi-output Gaussian process priors with realizations in the solution set of such systems, in particular only such solutions can be represented by Gaussian process regression. The construction is fully algorithmic via Gröbner bases and it does not employ any approximation. It builds these priors combining two parametrizations via a pullback: the first parametrizes the solutions for the system of differential equations and the second parametrizes all functions adhering to the boundary conditions.

1 Introduction

Gaussian processes (Rasmussen and Williams, 2006) are very data efficient. Hence, they are the prime regression technique for small datasets and applied when data is rare or expensive to produce. Applications range from robotics (Lima et al., 2018), biology (Honkela et al., 2015), global optimization (Osborne et al., 2009), anomaly detection Berns. et al. (2020), hyperparameter search Thornton et al. (2013) to engineering (Thewes et al., 2015). A Gaussian process can be viewed as a suitable probability distribution on a set of functions, which we can condition on observations using Bayes' rule. This avoids overfitting. Due to the self-conjugacy of the Gaussian distribution, the posterior is again Gaussian. The mean function of the posterior is used for regression and the variance quantifies uncertainty. For a suitable covariance function of the prior, the posterior can approximate any behavior present in data, even in noisy or unstructured data.

Any prior knowledge about the regression problem should be incorporated into the prior. Then, the preciously rare measurement data can be used to refine and improve on this prior knowledge, instead of needing to relearn it. The prior knowledge is usually encoded into the covariance structure of the Gaussian process, cf. (Rasmussen and Williams, 2006, §4) or (Duvenaud, 2014). Gaussian process regression differs in philosophy from deep learning, where the latter thrives on extracting knowledge from a lot of data but struggles with one-shot learning and encoding prior knowledge, which is usually done via pretraining on similar data.

Prior knowledge is often given by physical laws. In particular, it is important to include linear differential equations into machine learning frameworks. Gaussian processes that adhere to such a set of linear differential equations were constructed several times in the literature (Graepel, 2003; Macêdo and Castro, 2008; Särkkä, 2011; Scheuerer and Schlather, 2012; Wahlström et al., 2013; Solin et al., 2018; Jidling et al., 2017; Raissi et al., 2017; Raissi and Karniadakis, 2018; Jidling et al., 2018). All realizations and the mean function of the posterior strictly [1] satisfy these physical laws. Such Gaussian processes exist if and only if the set of linear differential equations describes a controllable system[2]. Their construction can be completely automatized by symbolic algorithms from algebraic system theory, which again strongly build on Gröbner bases (Lange-Hegermann, 2018).

While the above approaches are exact, there are also

[1] For notational simplicity, we refrain from using the phrase "almost surely" in this paper, e.g. by assuming separability.

[2] Controllable systems have "big" sets of solutions, even after adding boundary conditions. As there is no unique solution, one does regression or control in those systems, instead of (numerically) solving them as is usually done for systems with "small" solution sets.

Proceedings of the 24th International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130.

approximate approaches to include partial differential equations in Gaussian process and more generally machine learning. For example, various forms of posterior regularization (Ganchev et al., 2010; Song et al., 2016; Yuan et al., 2020) can flexibly consider any differential equation. The paper Raissi et al. (2018) constructed Gaussian processes on numerical difference approximation schemes of differential equations. Gaussian processes have been used to estimate conservation laws (Raissi and Karniadakis, 2017; Nguyen and Peraire, 2015, 2016). In (Yang et al., 2018), a Gaussian process prior is approximated from an MCMC scheme build on numerical simulations.

Usually, differential equations come with boundary conditions. Hence, a description of boundary conditions in a machine learning framework is highly desirable. In the special case of ODEs, boundary conditions behave as data points, hence one only needs finite-dimensional data to specify them. These data points can be trivially included into a Gaussian process (Kocijan et al., 2004; Calderhead et al., 2009; Barber and Wang, 2014; John et al., 2019) and other machine learning methods (Chen et al., 2018; Raissi, 2018; Särkkä and Solin, 2019). This paper claims no originality for ODEs.

For boundary conditions of PDEs, one would need functions (specified by infinite dimensional data) to describe the boundary conditions. Solving this problem *exactly* and without any approximation is the main contribution of this paper: the construction of (non-stationary) Gaussian process priors combining differential equations and general boundary conditions. This construction is again based on symbolically building parametrizations using Gröbner bases, as in (Lange-Hegermann, 2018).

More precisely, given a system of linear differential equations with rational (or, as a special case, constant) coefficient of a controllable system defined by an operator matrix and boundary conditions defined by the zero set of a polynomial ideal, we construct a Gaussian process prior of the corresponding set of smooth solutions. In particular, a regression model constructed from this Gaussian process prior has *only* solutions of the system as realizations. We need no approximations.

Using the results of this paper, one can add information to Gaussian processes by

(i) conditioning on data points (Bayes' rule),

(ii) restricting to solutions of linear operator matrices Lange-Hegermann (2018), and

(iii) adding boundary conditions (this paper).

Since these constructions are compatible, we can combine *strict, global information* from equations and boundary conditions with *noisy, local information* from observations. This paper is an example in how symbolic techniques can help data driven machine learning. All results are mathematically proven in the appendices and the algorithms are demonstrated on toy examples with only one or two data points, an extreme form of one-shot learning. The code for reproduction of the results is given in the appendix and the (very small amount of) data is completely given in the text of this paper.

The novelty in this paper does not lie in either of its techniques, which are well-known either in algebraic system theory or machine learning. Rather, this paper combines these techniques and thereby presents a novel framework to deal with learning from data in the presence of linear controllable *partial* differential equations and boundary conditions. We found it hard to compare to the state of the art, as there currently is no comparable technique, except the superficially similar paper (Graepel, 2003) discussed in Remark 5.5 and a plethora of machine learning techniques designed for *ordinary* differential equations. The only exception is Gulian et al. (2020), which considers inhomogeneous linear differential equations with boundary conditions using the spectral decomposition of a covariance function (Solin and Kok, 2019), where the right hand side is specified approximately by data. These approaches allow to approximately specify a prior for the solution of the differential equation, instead of specifying the prior for the parametrizing function as in this paper.

We recall Gaussian processes and their connection to linear operators in Section 2 and summarize the construction of Gaussian processes adhering to linear operators in Section 3. Describing boundary conditions as parametrizations is surprisingly simple (Section 4). Theorem 5.2 describes the core construction of this paper, which allows to check whether and how two parametrizations are combinable. In Section 6 we construct boundary conditions with non-zero right hand sides using the fundamental theorem on homomorphisms.

2 Operators and Gaussian Processes

A *Gaussian process* $g = \mathcal{GP}(\mu, k)$ is a probability distribution on the evaluations of functions $\mathbb{R}^d \to \mathbb{R}^\ell$ such that function values $g(x_1), \dots, g(x_n)$ are jointly Gaussian. It is specified by a *mean function* $\mu : \mathbb{R}^d \to \mathbb{R}^\ell : x \mapsto E(g(x))$ and a positive semidefinite *covariance function*

$$k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^{\ell\times\ell}_{\succeq 0} : \\ (x, x') \mapsto E\left((g(x) - \mu(x))(g(x') - \mu(x'))^T\right) .$$

All higher moments exists and are uniquely determined by μ and k, all higher cumulants are zero. We often restrict the domain of a Gaussian process to a subset of $\mathbb{R}^d$.

Assume the regression model $y_i = g(x_i)$ and condition on observations $(x_i, y_i) \in \mathbb{R}^{1\times d} \times \mathbb{R}^{1\times \ell}$ for $i = 1, \dots, n$. Denote by $k(x, X) \in \mathbb{R}^{\ell\times \ell n}$ resp. $k(X, X) \in \mathbb{R}^{\ell n\times \ell n}_{\succeq 0}$ the (covariance) matrices obtained by concatenating the matrices $k(x, x_j)$ resp. the positive semidefinite block partitioned matrix with blocks $k(x_i, x_j)$. Write $\mu(X)$ resp. $y \in \mathbb{R}^{1\times \ell n}$ for the row vector obtained by concatenating the rows $\mu(x_i)$ resp. y_i. The posterior is the Gaussian process

$$\mathcal{GP}\Big(\mu(x) + (y - \mu(X))k(X, X)^{-1}k(x, X)^T,\\ k(x, x') - k(x, X)k(X, X)^{-1}k(x', X)^T\Big).$$

Its mean function can be used as regression model and its variance as model uncertainty.

Gaussian processes are the linear objects among stochastic processes and their rich connection with linear operators is present everywhere in this paper. In particular, the class of Gaussian processes is closed under linear operators once mild assumptions hold. Now, we formalize and generalize the following well-known example of differentiating a Gaussian process.

Example 2.1. Let $g = \mathcal{GP}(0, k(x, x'))$ be a scalar univariate Gaussian process with differentiable realizations. Then,

$$\left[\tfrac{\partial}{\partial x}\right]_* g := \mathcal{GP}\left(0, \frac{\partial^2}{\partial x \partial x'} k(x, x')\right)$$

is the Gaussian process of derivatives of realizations of the Gaussian process g. One can interpret this Gaussian process $\left[\tfrac{\partial}{\partial x}\right]_* g$ as taking derivatives as measurement data and producing a regression model of derivatives. Taking a one-sided derivative $\frac{\partial}{\partial x}k(x, x')$ yields the cross-covariance between a function and its derivative. See (Cramér and Leadbetter, 2004, §5.2) for a proof and (Wu et al., 2017) resp. Cobb et al. (2018) for a applications in Bayesian optimization resp. vector field modeling.

Given a set of functions $G \subseteq \{f : X \to Y\}$ and $b : Y \to Z$, then the *pushforward* is

$$b_*G = \{b \circ f \mid f \in G\} \subseteq \{f : X \to Z\}.$$

$$X \xrightarrow{G} Y \xrightarrow{b} Z, \quad X \xrightarrow{b_*G} Z$$

A *pushforward* of a stochastic Process $g : \Omega \to (X \to Y)$ by $b : Y \to Z$ is

$$b_*g : \Omega \to (X \to Z) : \omega \mapsto (b \circ g(\omega)).$$

Lemma 2.2. *Let $\mathcal{F}$ and $\mathcal{G}$ be spaces of functions defined on $X \subseteq \mathbb{R}^d$ with product σ-algebra of function evalutions. Let $g = \mathcal{GP}(\mu(x), k(x, x'))$ with realizations in $\mathcal{F}$ and $B : \mathcal{F} \to \mathcal{G}$ a linear, measurable operator which commutes with expectation w.r.t. the measure induced by g on $\mathcal{F}$ and by B_*g on $\mathcal{G}$. Then, the pushforward B_*g of g under B is again Gaussian with*

$$B_*g = \mathcal{GP}(B\mu(x), Bk(x, x')(B')^T) \ ,$$

where B' denotes the operation of B on functions with argument x'.

Call B_*g the *pushforward Gaussian process* of g under B.

We postpone the proof to the appendix. Lemma 2.2 is often stated without assuming that B commutes with expectation, but also without proof. If such a more general version of Lemma 2.2 holds, the author would be very interested to see a reference. Special cases have been discussed in the literature, often only for mean square differentiability (Papoulis and Pillai, 2002, after (9.87) resp. (10.78); in the first and second resp. third edition; Adler, 1981, Thm 2.2.2, Agrell, 2019; Da Veiga and Marrel, 2012, §2.3; Bertinet and Agnan, 2004, Thm. 9).

Consider change points and change surfaces as application of Lemma 2.2, following (Garnett et al., 2009, 2010; Lloyd et al., 2014; Herlands et al., 2016).

Example 2.3. Let $\rho_1, \rho_2 : \mathbb{R}^d \to [0, 1]$ a *partition of unity*, i.e., $\rho_1(x) + \rho_2(x) = 1$ for all $x \in \mathbb{R}^d$. Usually, both ρ_1 and ρ_2 are close to being 0 or close to being 1 over most of $\mathbb{R}^d$. Such a partition of unity induces a linear operator

$$\rho = \begin{bmatrix}\rho_1 & \rho_2\end{bmatrix} : \mathcal{F}^{2\times 1} \to \mathcal{F}^{1\times 1} : \begin{bmatrix}f_1\\ f_2\end{bmatrix} \mapsto \begin{bmatrix}\rho_1 & \rho_2\end{bmatrix}\begin{bmatrix}f_1\\ f_2\end{bmatrix},$$

where $\mathcal{F}$ is a space of functions $\mathbb{R}^d \to \mathbb{R}$. Given two independent Gaussian processes $g_1 = \mathcal{GP}(0, k_1)$, $g_2 = \mathcal{GP}(0, k_2)$ with realizations in $\mathcal{F}$, we have

$$\begin{aligned}\rho_* g &:= \left[\begin{bmatrix}\rho_1 & \rho_2\end{bmatrix}_*\begin{bmatrix}g_1\\ g_2\end{bmatrix}\right.\\ &= \mathcal{GP}\left(0, \rho_1(x)k_1(x, x')\rho_1(x') + \rho_2(x)k_2(x, x')\rho_2(x')\right)\end{aligned}$$

Thereby, we model change points (for $d = 1$) or change surfaces (for $d > 1$) at positions where ρ_1 changes from being close to 0 to being close to 1. This example is the basis for boundary conditions in Section 4: when setting g_2 to zero, $\rho_* g$ is close to zero where $\rho_2 \approx 1$ and close to g_1 where $\rho_1 \approx 1$.

3 Solution Sets of Operator Equations

We consider linear ordinary and partial differential equations defined on the set of smooth functions. Let

$\mathcal{F} = C^\infty(X, \mathbb{R})$ be the real vector space of smooth functions from $X \subseteq \mathbb{R}^d$ to $\mathbb{R}$ with the usual Fréchet topology[3]. The *squared exponential covariance function*

$$k_{\mathcal{F}}(x_i, x_j) = \exp\left(-\frac{1}{2}\sum_{a=1}^{d}(x_{i,a} - x_{j,a})^2\right) \quad (1)$$

induces a Gaussian process prior $g_{\mathcal{F}} = \mathcal{GP}(0, k_{\mathcal{F}})$ with realizations dense in the space of smooth functions $\mathcal{F} = C^\infty(X, \mathbb{R})$ w.r.t. this topology.

The following three *rings of linear operators* R model operator equations. These rings are $\mathbb{R}$-algebras s.t. $\mathcal{F}$ is a *topological R-(left-)module*, i.e., $\mathcal{F}$ is a topological $\mathbb{R}$-vector space of functions $X \to \mathbb{R}$ for $X \subseteq \mathbb{R}^d$ that also is an R-(left-)module such that the elements of R operate continuously on $\mathcal{F}$.

Example 3.1. The polynomial[4] ring $R = \mathbb{R}[\partial_{x_1}, \ldots, \partial_{x_d}]$ models linear differential equations with constant coefficients, as ∂_{x_i} acts on $\mathcal{F} = C^\infty(X, \mathbb{R})$ via partial derivative w.r.t. x_i.

Example 3.2. The polynomial ring $R = \mathbb{R}[x_1, \ldots, x_d]$ models algebraic equations via multiplication on $\mathcal{F}$. This ring is relevant for boundary conditions.

To combine linear differential equations with constant coefficients with boundary conditions or to model linear differential equations with polynomial[5] coefficients, consider the following ring.

Example 3.3. The Weyl algebra $R = \mathbb{R}[x_1, \ldots, x_n]\langle\partial_{x_1}, \ldots, \partial_{x_n}\rangle$ has the non-commutative relation $\partial_{x_i} x_j = x_j \partial_{x_i} + \delta_{ij}$ representing the product rule of differentiation, where δ_{ij} is the Kronecker delta.

Operators defined over these three rings satisfy the assumptions of Lemma 2.2: multiplication commutes with expectations and the dominated convergence theorem implies that expectation commutes with derivatives, as realizations of $g_{\mathcal{F}}$ are continuously differentiable. These three rings also operate continuously on $\mathcal{F}$: the Fréchet topology is constructed to make derivation continuous, and multiplication is bounded (if X is bounded and bounded away from infinity) and hence continuous, as $\mathcal{F}$ is Fréchet.

[3]For Gaussian processes on Fréchet spaces see (Zapała, 2002; Osswald, 2012). The topology is generated by the separating family $\|f\|_{a,b} := \sup_{i \in \mathbb{Z}^d_{\geq 0}, |i| \leq a} \sup_{z \in [-b,b]^d} |\frac{\partial}{\partial z^i} f(z)|$ of seminorms for $a, b \in \bar{\mathbb{Z}}_{\geq 0}$ on $\mathcal{F}$ (Treves, 1967, §10).

[4]Partial derivatives commute (symmetry of 2nd derivatives) and generate a *commutative* polynomial ring.

[5]No major changes for rational, holonomic, or meromorphic coefficients.

3.1 Parametrizations

For $A \in R^{\ell' \times \ell}$ define the *solution set* $\mathrm{sol}_{\mathcal{F}}(A) := \{f \in \mathcal{F}^{\ell \times 1} \mid Af = 0\}$ as a nullspace of an operator matrix A. We say that a Gaussian process is *in* a function space, if its realizations are contained in said space. The following tautological lemma is a version of the fundamental theorem of homomorphisms. It describes the interplay of Gaussian processes and solution sets of operators.

Lemma 3.4 (Lange-Hegermann, 2018, Lemma 2.2)**.** *Let $g = \mathcal{GP}(\mu, k)$ be a Gaussian process in $\mathcal{F}^{\ell \times 1}$. Then g is a Gaussian process in the solution set $\mathrm{sol}_{\mathcal{F}}(A)$ of $A \in R^{\ell' \times \ell}$ if and only if both μ is contained in $\mathrm{sol}_{\mathcal{F}}(A)$ and $A_*(g - \mu)$ is the constant zero process.*

To construct Gaussian processes with realizations in the solution set $\mathrm{sol}_{\mathcal{F}}(A)$ of an operator matrix $A \in R^{\ell' \times \ell}$, one looks for a $B \in R^{\ell \times \ell''}$ with $AB = 0$ (Jidling et al., 2017). If $g = \mathcal{GP}(0, k)$ is a Gaussian process in $\mathcal{F}^{\ell'' \times 1}$, then the realizations of $B_* g$ are contained in $\mathrm{sol}_{\mathcal{F}}(A)$ by Lemma 3.4, as $A_*(B_* g) = (AB)_* g = 0_* g = 0$. In practice, one would like that any solution in $\mathrm{sol}_{\mathcal{F}}(A)$ can be approximated by $B_* g$ to arbitrary precision, i.e., that the realizations of the Gaussian process $B_* g$ are dense in $\mathrm{sol}_{\mathcal{F}}(A)$. To this end, we call $B \in R^{\ell \times \ell''}$ a *parametrization* of $\mathrm{sol}_{\mathcal{F}}(A)$ if $\mathrm{sol}_{\mathcal{F}}(A) = B\mathcal{F}^{\ell'' \times 1}$.

Proposition 3.5 (Lange-Hegermann, 2018, Prop. 2.4)**.** *Let $B \in R^{\ell \times \ell''}$ be a parametrization of $\mathrm{sol}_{\mathcal{F}}(A)$ for $A \in R^{\ell' \times \ell}$. Take the Gaussian process $g_{\mathcal{F}}^{\ell'' \times 1}$ of ℓ'' i.i.d. copies of $g_{\mathcal{F}}$, the Gaussian process with squared exponential covariance[6] function $k_{\mathcal{F}}$ from Eq.* (1)*. Then, the realizations of $B_* g_{\mathcal{F}}^{\ell'' \times 1}$ are dense in $\mathrm{sol}_{\mathcal{F}}(A)$.*

Proof. By construction, realizations of $g_{\mathcal{F}}^{\ell'' \times 1}$ are dense in $\mathcal{F}^{\ell'' \times 1}$. The operator B induces a surjective continuous ($\mathcal{F}$ is a topological R-module) map. Surjective continuous maps map dense sets to dense sets. □

3.2 Algorithmically constructing parametrizations

We summarize the algorithm which decides whether a parametrization of a system of linear differential equations exists and compute it in the positive case. To construct the parametrization B, we are lead to just compute the nullspace[7] of $\mathcal{F}^{\ell' \times 1} \xleftarrow{A} \mathcal{F}^{\ell \times 1}$. This is not feasible, as $\mathcal{F}$ is too "big" to allow computations. Instead, we compute the nullspace of $R^{\ell' \times 1} \xleftarrow{A} R^{\ell \times 1}$,

[6]Or any other covariance with realizations dense in $\mathcal{F}$.

[7]We avoid calling nullspaces kernel, due to confusion with symmetric positive semidefinite functions. While a left resp. right nullspace is a module, we abuse notation and denote any matrix as left resp. right nullspace if its rows resp. columns generate the nullspace as an R-module.

a symbolic computation, only using operations over R without involvement of $\mathcal{F}$.

Theorem 3.6. *Let $A \in R^{\ell' \times \ell}$. Let B be the right nullspace of A and A' the left nullspace of B. Then $\mathrm{sol}_{\mathcal{F}}(A')$ is the largest subset of $\mathrm{sol}_{\mathcal{F}}(A)$ that is parametrizable and B parametrizes $\mathrm{sol}_{\mathcal{F}}(A')$.*

A well-known and trivial special case of this theorem are linear equations in finite dimensional vector spaces, with $R = \mathcal{F} = \mathbb{R}$ the field of real numbers. In that case, $\mathrm{sol}_{\mathcal{F}}(A)$ can be found by applying the Gaussian algorithm to the homogeneous system of linear equations $Ab = 0$ and write a base for the solutions of b as columns of a matrix B. This matrix B is the (right) nullspace of A. There are no additional equations satisfied by the above solutions, i.e. $A = A'$ generates the (left) nullspace of B.

In general, the left nullspace A' of the right nullspace B of A is not necessarily A. E.g., for the univariate polynomial ring $R = \mathbb{R}[x]$ and the matrix $A = \begin{bmatrix} x \end{bmatrix}$ we have $B = \begin{bmatrix} 0 \end{bmatrix}$ and $A' = \begin{bmatrix} 1 \end{bmatrix}$.

Corollary 3.7. *In Theorem 3.6, $\mathrm{sol}_{\mathcal{F}}(A)$ is parametrizable if and only if the rows of A and A' generate the same row-module. Since $AB = 0$, this is the case if all rows of A' are contained in the row module generated by the rows of A. In this case, $\mathrm{sol}_{\mathcal{F}}(A)$ is parametrized by B.*

For a formal proof we refer to the literature (Zerz et al., 2010, Thm. 2; Zerz, 2000, Thm. 3, Alg. 1, Lemma 1.2.3; Oberst, 1990, §7.(24); Quadrat, 2013, 2010; Barakat, 2010; Seiler and Zerz, 2010; Chyzak et al., 2005; Robertz, 2015). Luckily, there is a high level description of the parametrizable systems.

Theorem 3.8 (Oberst, 1990, §7.(21))**.** *A system $\mathrm{sol}_{\mathcal{F}}(A)$ is parametrizable iff it is controllable.*

The intuition for controllability is that one can partition the functions of the system into state and input, such that any chosen state can be reached by suitably manipulating the inputs. In particular, controllable systems (except the trivial system) are far away from having a unique solution. If A is not parametrizable, then the solution set $\mathrm{sol}_{\mathcal{F}}(A')$ is the subset of controllable behaviors in $\mathrm{sol}_{\mathcal{F}}(A)$.

Reduced Gröbner bases generalize the reduced echelon form from linear systems to systems of polynomial (and hence linear operator) equations, by bringing them into a standard form. They are computed by Buchberger's algorithm, which is a generalization of the Gaussian and Euclidean algorithm and a special case of the Knuth-Bendix completion algorithm for rewriting systems. The generalization of Gröbner bases to vectors of polynomials is straight forward.

Gröbner bases make the above theorems algorithmic. Similar to the reduced echelon form, Gröbner bases allow to compute all solutions over R of the homogeneous system and compute, if it exists, a particular solution over R for an inhomogeneous system. Solving homogeneous systems is the same as computing its right resp. left nullspace (of A resp. B). Solving inhomogeneous equations decides whether an element (the rows of A') is contained in a module (the row module of A). A formal description of Gröbner bases exceeds the scope of this note. We refer to the excellent literature (Sturmfels, 2005; Eisenbud, 1995; Adams and Loustaunau, 1994; Greuel and Pfister, 2008; Gerdt, 2005; Buchberger, 2006). Not only do they generalize the Gaussian algorithm for linear polynomials, but also the Euclidean algorithm for univariate polynomials. In addition to polynomial rings, Gröbner bases also exist for the Weyl algebra (Robertz, 2006, 2008; Chyzak et al., 2007; Levandovskyy, 2005; Levandovskyy and Schönemann, 2003) and many further rings. The algorithms used in the paper are usually readily available functions implemented in various computer algebra systems (Decker et al., 2019; Grayson and Stillman, 1992). While Gröbner bases depend on the choice of a term order, similar to reordering columns in the Gaussian algorithm, any term order leads to correct results.

Gröbner bases solve problems of high complexity like ExpSpace completeness (Mayr, 1989; Mayr and Meyer, 1982; Bayer and Stillman, 1988). In practice, this is less of a problem, as the Gröbner basis computations only involve the operator equations, but no data. Hence we view the complexity of the Gröbner basis computations in $\mathcal{O}(1)$, which only needs to be applied once to construct the covariance function. In particular, the Gröbner bases of every example in this paper terminate instantaneously. For larger examples, the data dependent $\mathcal{O}(n^3)$ of the Gaussian processes is the computationally restricting subalgorithm.

Example 3.9 (Lange-Hegermann, 2018, Example 4.4)**.** We construct a prior for smooth tangent fields on the sphere without sources and sinks using the polynomial Weyl algebra $R = \mathbb{R}[x, y, z]\langle \partial_x, \partial_y, \partial_z \rangle$. I.e., we are interested in $\mathrm{sol}_A(\mathcal{F}) = \{v \in C^\infty(S^2, \mathbb{R}^3) \mid Av = 0\}$ for

$$A := \begin{bmatrix} x & y & z \\ \partial_x & \partial_y & \partial_z \end{bmatrix}.$$

The right nullspace

$$B := \begin{bmatrix} -z\partial_y + y\partial_z \\ z\partial_x - x\partial_z \\ -y\partial_x + x\partial_y \end{bmatrix}.$$

can be checked to yield a parametrization of $\mathrm{sol}_{\mathcal{F}}(A)$. For a demonstration of this covariance functions, see Figure 1.

4 Boundary conditions

Differential equations and boundary conditions go hand in hand in applications. Here, we recall a general methods to incorporate boundary conditions into Gaussian processes, a slight generalization of (Graepel, 2003, Section 3), closely related to vertical rescaling. Boundary conditions in ODEs are equivalent to conditioning on data points John et al. (2019).

We recall the creation of priors for homogeneous boundary conditions for PDEs from Graepel (2003), for the inhomogeneous case see Section 6. Such boundary conditions fix the function values and/or their derivatives at a subset of the domain X exactly. We restrict ourselves to zero sets of polynomials. For more complicated, approximate boundary conditions see Solin and Kok (2019) and for asymptotic boundaries see Tan (2018).

Denote again by $\mathcal{F} = C^\infty(X, \mathbb{R})$ the set of smooth functions defined on $X \subset \mathbb{R}^d$ compact. Let $R' \subset \mathbb{R}^X$ be a Noetherian ring of functions and subring of R, and $M \subseteq X$ implicitely defined

$$M = \mathcal{V}(I) := \{m \in X \mid f(m) = 0 \text{ for all } f \in I\}$$

for an ideal $I \trianglelefteq R'$ of equations. An important example for this setting is the Weyl algebra $R = \mathbb{R}[x_1, \ldots, x_d]\langle \partial_{x_1}, \ldots, \partial_{x_d}\rangle$ and its subring R' the polynomial ring $R' = \mathbb{R}[x_1, \ldots, x_d]$.

Proposition 4.1. *A row $B' = \begin{bmatrix} f_1 & \ldots & f_{\ell''} \end{bmatrix}$ whose entries generate the ideal I parametrizes all solutions of a homogenous boundary condition $f_{|M} = 0$ for a function $f \in \mathcal{F}$ via $\mathcal{F} \xleftarrow{B'} \mathcal{F}^{\ell'' \times 1}$*

Proof. Let on the one hand $p \in \mathbb{R}^d$ such that $f_i(p) = 0$ for all $1 \leq i \leq \ell''$. Then, $(B'g)(p) = 0$ for all $g \in \mathcal{F}^{\ell'' \times 1}$. On the other hand, let $p \in \mathbb{R}^d$ such that there is an $1 \leq j \leq \ell''$ with $f_j(p) \neq 0$ and parametrize $h \in \mathcal{F}$ locally as $h(x) = B' \cdot \frac{h(x)}{f_j(x)} e_j$ for e_j the jth standard basis vector, since locally $f_j(x) \neq 0$. For a global parametrization, patch the local parametrizations via a partition of unity. □

To encode boundary conditions for $\ell > 1$ functions, we use a direct sum matrix $B' \in (R')^{\ell \times \ell\ell''}$, e.g., $B' = \begin{bmatrix} B'_1 & 0 \\ 0 & B'_2 \end{bmatrix}$ for $\ell = 2$ where B'_1 and B'_2 are rows over R' describing the boundaries.

Example 4.2. Functions $\mathcal{F} = C^\infty([0,1]^2, \mathbb{R})$ with Dirichlet boundary conditions $f(0,y) = f(1,y) = f(x,0) = f(x,1) = 0$ are parametrized by $B' = \begin{bmatrix} x(x-1)y(y-1) \end{bmatrix}$.

Example 4.3. Functions $\mathcal{F} = C^\infty(\mathbb{R}^3, \mathbb{R})$ with boundary condition $f(0,0,z) = 0$ are parametrized by $B' = \begin{bmatrix} x & y \end{bmatrix}$.

Example 4.4. Consider $\mathcal{F} = C^\infty(\mathbb{R}^2, \mathbb{R})$ with boundary conditions $f(0,y) = \left(\frac{\partial}{\partial x} f(x,y)\right)_{|x=0} = 0$. Such functions are parametrized by $B = \begin{bmatrix} x^2 \end{bmatrix}$, since

$$\left(\frac{\partial}{\partial x}(x^2 f(x,y))\right)_{|x=0} = \left(2xf(x,y) + x^2 \frac{\partial}{\partial x} f(x,y)\right)_{|x=0} = 0.$$

5 Intersecting parametrizations

Now, we combine parametrizations $B_1 \in R^{\ell \times \ell''}$ and $B_2 \in R^{\ell \times \ell'''}$, e.g. from differential equations and boundary conditions, by intersecting their images $B_1\mathcal{F}^{\ell''} \cap B_2\mathcal{F}^{\ell'''}$.

Example 5.1. Actually, the Dirichlet boundary condition of Example 4.2 is an intersection of the images of the boundary conditions parametrized by $\begin{bmatrix} x \end{bmatrix}$, $\begin{bmatrix} x-1 \end{bmatrix}$, $\begin{bmatrix} y \end{bmatrix}$, and $\begin{bmatrix} y-1 \end{bmatrix}$.

The following theorem is the main contribution of this paper. It constructs a parametrization of intersections of parametrizations algorithmically.

Theorem 5.2 (Intersecting parametrizations)**.** *Let $B_1 \in R^{\ell \times \ell''_1}$ and $B_2 \in R^{\ell \times \ell''_2}$. Denote by*

$$C := \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} \in R^{(\ell''_1 + \ell''_2) \times m}$$

the right-nullspace of the matrix $B := \begin{bmatrix} B_1 & B_2 \end{bmatrix} \in R^{\ell \times (\ell''_1 + \ell''_2)}$. Then $B_1C_1 = -B_2C_2$ parametrizes solutions of $B_1\mathcal{F}^{\ell''_1} \cap B_2\mathcal{F}^{\ell''_2}$.

For the proof cf. the appendix. The computations are again Gröbner basis computations over the ring R.

Example 5.3. We rephrase the computation of divergence free fields on the sphere from Example 3.9. This is the intersection of divergence free fields, the zero set of $A_1 := \begin{bmatrix} \partial_x & \partial_y & \partial_z \end{bmatrix}$, and the fields on the sphere, the zero set of $A_2 := \begin{bmatrix} x & y & z \end{bmatrix}$, respectively parametrized by

$$B_1 = \begin{bmatrix} 0 & \partial_z & -\partial_y \\ -\partial_z & 0 & \partial_x \\ \partial_y & -\partial_x & 0 \end{bmatrix} \text{ and } B_2 = \begin{bmatrix} 0 & z & -y \\ -z & 0 & x \\ y & -x & 0 \end{bmatrix}.$$

The right-nullspace of $\begin{bmatrix} B_1 & B_2 \end{bmatrix}$ is

$$C = \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} = \begin{bmatrix} x & \partial_x & 0 \\ y & \partial_y & 0 \\ z & \partial_z & 0 \\ \partial_x & 0 & x \\ \partial_y & 0 & y \\ \partial_z & 0 & z \end{bmatrix}$$

The matrix $\begin{bmatrix} B_1 & B_2 \end{bmatrix}$ is the left nullspace of C. Now,

$$B_1C_1 = -B_2C_2 = \begin{bmatrix} z\partial_y - y\partial_z & 0 & 0 \\ -z\partial_x + x\partial_z & 0 & 0 \\ y\partial_x - x\partial_y & 0 & 0 \end{bmatrix}$$

is equivalent[8] to the matrix B from Example 3.9.

Example 5.4. We continue with the divergence free fields on the sphere from Examples 3.9 and 5.3. These are parametrized by

$$B_1 := \begin{bmatrix} -z\partial_y + y\partial_z \\ z\partial_x - x\partial_z \\ -y\partial_x + x\partial_y \end{bmatrix}.$$

functions vanishing at the equator (boundary condition: $f(x, y, 0) = 0$) are parametrized by

$$B_2 := \begin{bmatrix} z & 0 & 0 \\ 0 & z & 0 \\ 0 & 0 & z \end{bmatrix}.$$

The nullspace of $\begin{bmatrix} B_1 & B_2 \end{bmatrix}$ is

$$C := \begin{bmatrix} C_1 \\ C_{2,1} \\ C_{2,2} \\ C_{2,3} \end{bmatrix} = \begin{bmatrix} -z^2 \\ z^2\partial_y - yz\partial_z - 2y \\ -z^2\partial_x + xz\partial_z + 2\,x \\ yz\partial_x - xz\partial_y \end{bmatrix}.$$

The left nullspace of C is not only generated by $\begin{bmatrix} B_1 & B_2 \end{bmatrix}$, but by the additional relation $D := \begin{bmatrix} 0 & x & y & z \end{bmatrix}$. This relation D tells us, that the parametrized solutions of C_2 are a vector field on a sphere around the origin, which they remain after being multiplied by the scalar matrix B_2. We gladly accept this additional condition. Now,

$$B_1C_1 = -B_2C_2 = \begin{bmatrix} -z^3\partial_y + yz^2\partial_z + 2yz \\ z^3\partial_x - xz^2\partial_z - 2xz \\ -yz^2\partial_x + xz^2\partial_y \end{bmatrix}$$

parametrizes the divergence free fields on the sphere vanishing at the equator, see Figure 1.

Remark 5.5. (Graepel, 2003) also constructs a Gaussian process prior for a system $Af = y$ of linear differential equations with boundary conditions. It assumes any Gaussian process prior on f and uses a variant of Lemma 2.2 to compute the cross-covariance between y and f, which allows to condition the model $p(f)$ on data for y. This model ensures in no way that f is constrained to solutions of $Af = y$, even if e.g. $y = 0$ is known. Furthermore, conditioning $p(f)$ on data for f is just done w.r.t. the (uninformative) Gaussian process prior chosen for f.

[8] The matrices B_1 and B_2 each have a non-zero nullspace, corresponding to the two trivial columns in B_1C_1.

As in this paper, (Graepel, 2003) uses Proposition 4.1 and a pushforward to construct a prior for f supported on solutions of the homogeneous boundary condition. No effort to combine differential equations and boundary conditions as in Theorem 5.2 is necessary, since the differential equations are not satisfied anyway. The case of inhomogeneous boundary conditions is solved via taking a particular solutions as a mean function. Finding such a particular solution is simple, as only the boundary conditions must be satisfied; in contrast to Section 6 of this paper, where also the differential equations need to be satisfied.

6 Inhomogenous boundary conditions

So far, we have only considered homogeneous equations and boundary conditions, i.e., with right hand sides zero. The fundamental theorem of homomorphisms (cf. Lemma 3.4) extends this to the inhomogeneous case, by taking a particular solution as mean function. While simple theoretically, finding a particular solution can be quite hard in practice. We restrict ourselves to examples.

Example 6.1. Consider smooth divergence free fields on the 2-sphere $X = S^2$, i.e., $f \in \mathcal{F}^{3\times 1}$ with

$$Af = \begin{bmatrix} x & y & z \\ \partial_x & \partial_y & \partial_z \end{bmatrix} f = 0$$

and inhomogeneous boundary condition $f_3(x, y, 0) = y$.

The function $\mu = \begin{bmatrix} 0 & -z & y \end{bmatrix}^T$ is a particular solution. Hence, we take it as mean function. The matrix $B_1C_1 = -B_2C_2$ from Example 5.4 parametrizes functions with the corresponding homogeneous boundary condition $f_3(x, y, 0) = 0$ of functions vanishing at the equator.

Hence, assuming mean zero and squared exponential covariance $k_{\mathcal{F}}$, the Gaussian process $\mathcal{GP}\left(\mu, (B_1C_1)k_{\mathcal{F}}((B_1C_1)')^T\right)$ is a prior distribution in the solutions of the equations and boundary conditions by Lemma 3.4, which we demonstrate in Figure 1.

Example 6.2. Consider smooth divergence free fields on the square $X = [0, 1] \times [0, 1]$ such that no flow in or out of X is possible at the lower and upper boundary of X and there is a constant flow of strength 1 in x-direction at the left and right boundary. The divergence-freeness is modelled by the right kernel

$$B_1 = \begin{bmatrix} \partial_y \\ -\partial_x \end{bmatrix}$$

of $A = \begin{bmatrix} \partial_x & \partial_y \end{bmatrix}$. We model the conditions on the flow by the constant mean function

$$\mu : (x, y) \mapsto \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

Figure 1: On the left, the posterior mean of conditioning the prior in Example 3.9 at two opposite points on the equator with tangent vectors pointing north. Without sources and sinks, the tangent vectors flow south away from the data. In the middle, the posterior mean from Example 5.4 of a divergence free tangent field on the sphere which is zero at the equator (red) and conditioned at a *single* observation at the north pole. Notice the flow parallel to the equator in middle latitudes, orthogonal to the observation, avoids sinks or sources. On the right, the posterior mean from Example 6.1 of a divergence free tangent field on the sphere with the given boundary condition (red) at the equator being conditioned at a single observation at the north pole. Data is displayed artificially bigger.

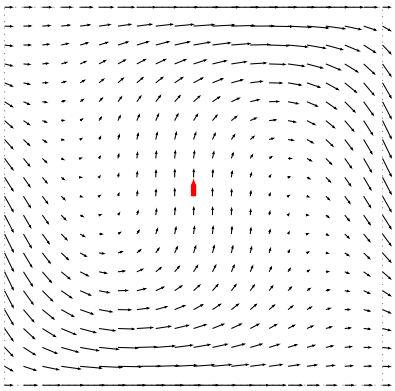

Figure 2: A plot of the model from Example 6.2, conditioned on the vector $(0, 1)$ at the point $(0.5, 0.5)$, which is plotted artificially bigger and red.

describing flow in x-direction and the boundary condition parametrized by

$$B_2 = \begin{bmatrix} x(x-1) & 0 \\ 0 & y(y-1) \end{bmatrix}.$$

The nullspace of $\begin{bmatrix} B_1 & B_2 \end{bmatrix}$ is

$$C := \begin{bmatrix} C_1 \\ C_{2,1} \\ C_{2,2} \end{bmatrix} = \begin{bmatrix} x^2y^2 - x^2y - xy^2 + xy \\ -y^2\partial_y + y\partial_y - 2y + 1 \\ x^2\partial_x - x\partial_x + 2x - 1 \end{bmatrix}.$$

and leads to the parametrization

$$\begin{aligned} P &:= B_1C_1 = -B_2C_2 \\ &= \begin{bmatrix} x(x-1)(-1 + y^2\partial_y + y(-\partial_y + 2)) \\ -y(y-1)(-1 + x^2\partial_x + x(-\partial_x + 2)) \end{bmatrix}. \end{aligned}$$

Hence, assuming a squared exponential covariance $k_{\mathcal{F}}$ for the parametrizing function the Gaussian process

$$\mathcal{GP}\left(\mu, Pk_{\mathcal{F}}P'^{T}\right)$$

is a prior of smooth divergence free fields on X with the above flow conditions. We demonstrate this prior in Figure 6.

7 Conclusion

This paper incorporates prior knowledge into machine learning frameworks. It presents a novel framework to

1. describe parametrizations for boundary conditions,
2. combine parametrizations by intersecting their images, and
3. build Gaussian process priors with realizations in the solution set of a system of linear differential equations with boundary conditions,

without any assumptions or approximations. These priors have been demonstrated on geometric problems and lead to reasonable models with one or two (cf. Figure 1) data points.

The author thanks the reviewers for their constructive feedback and by is interested in further work on encoding physical or system-theoretic properties in Gaussian process priors.

References

Adams, W. W. and Loustaunau, P. (1994). *An Introduction to Gröbner Bases*. Graduate Studies in Mathematics. American Mathematical Society.

Adler, R. (1981). *The Geometry of Random Fields*. Classics in Applied Mathematics. Society for Industrial and Applied Mathematics.

Agrell, C. (2019). Gaussian Processes with Linear Operator Inequality Constraints. *arXiv preprint arXiv:1901.03134*.

Barakat, M. (2010). Purity Filtration and the Fine Structure of Autonomy. In *Proceedings of the 19th International Symposium on Mathematical Theory of Networks and Systems - MTNS 2010*, pages 1657–1661, Budapest, Hungary.

Barber, D. and Wang, Y. (2014). Gaussian processes for bayesian estimation in ordinary differential equations. In *International conference on machine learning*, pages 1485–1493.

Bayer, D. and Stillman, M. (1988). On the Complexity of Computing Syzygies. *Journal of Symbolic Computation*, 6(2-3):135–147.

Berns., F., Lange-Hegermann., M., and Beecks., C. (2020). Towards gaussian processes for automatic and interpretable anomaly detection in industry 4.0. In *Proceedings of the International Conference on Innovative Intelligent Industrial Production and Logistics - Volume 1: IN4PL,*, pages 87–92. INSTICC, SciTePress.

Bertinet, A. and Agnan, T. C. (2004). *Reproducing Kernel Hilbert Spaces in Probability and Statistics*. Kluwer Academic Publishers.

Buchberger, B. (2006). An Algorithm for Finding the Basis Elements of the Residue Class Ring of a Zero Dimensional Polynomial Ideal. *J. Symbolic Comput.*, 41(3-4):475–511. Translated from the 1965 German original by Michael P. Abramson.

Calderhead, B., Girolami, M., and Lawrence, N. D. (2009). Accelerating bayesian inference over nonlinear differential equations with gaussian processes. In Koller, D., Schuurmans, D., Bengio, Y., and Bottou, L., editors, *Advances in Neural Information Processing Systems 21*, pages 217–224. Curran Associates, Inc.

Chen, T. Q., Rubanova, Y., Bettencourt, J., and Duvenaud, D. K. (2018). Neural ordinary differential equations. In *Advances in neural information processing systems*, pages 6571–6583.

Chyzak, F., Quadrat, A., and Robertz, D. (2005). Effective algorithms for parametrizing linear control systems over Ore algebras. *Appl. Algebra Engrg. Comm. Comput.*, 16(5):319–376.

Chyzak, F., Quadrat, A., and Robertz, D. (2007). OreModules: a Symbolic Package for the Study of Multidimensional Linear Systems. In *Applications of time delay systems*, volume 352 of *Lecture Notes in Control and Inform. Sci.*, pages 233–264. Springer, Berlin.

Cobb, A. D., Everett, R., Markham, A., and Roberts, S. J. (2018). Identifying sources and sinks in the presence of multiple agents with gaussian process vector calculus. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1254–1262.

Cramér, H. and Leadbetter, M. R. (2004). *Stationary and Related Stochastic Processes*. Sample function properties and their applications, Reprint of the 1967 original.

Da Veiga, S. and Marrel, A. (2012). Gaussian Process Modeling with Inequality Constraints. In *Annales de la Faculté des sciences de Toulouse: Mathématiques*, volume 21, pages 529–555.

Decker, W., Greuel, G.-M., Pfister, G., and Schönemann, H. (2019). SINGULAR 4-1-2 — A computer algebra system for polynomial computations. `http://www.singular.uni-kl.de`.

Duvenaud, D. (2014). *Automatic Model Construction with Gaussian Processes*. PhD thesis, University of Cambridge.

Eisenbud, D. (1995). *Commutative Algebra with a View Toward Algebraic Geometry*, volume 150 of *Graduate Texts in Mathematics*. Springer-Verlag.

Ganchev, K., Gillenwater, J., Taskar, B., et al. (2010). Posterior regularization for structured latent variable models. *Journal of Machine Learning Research*, 11(Jul):2001–2049.

Garnett, R., Osborne, M. A., Reece, S., Rogers, A., and Roberts, S. J. (2010). Sequential bayesian prediction in the presence of changepoints and faults. *The Computer Journal*, 53(9):1430–1446.

Garnett, R., Osborne, M. A., and Roberts, S. J. (2009). Sequential Bayesian Prediction in the Presence of Changepoints. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 345–352. ACM.

Gerdt, V. P. (2005). Involutive Algorithms for Computing Gröbner bases. In *Computational commutative and non-commutative algebraic geometry*, volume 196 of *NATO Sci. Ser. III Comput. Syst. Sci.*, pages 199–225.

Graepel, T. (2003). Solving Noisy Linear Operator Equations by Gaussian Processes: Application to Ordinary and Partial Differential Equations. In *Proceedings of the Twentieth International Conference*

on International Conference on Machine Learning, ICML'03, pages 234–241. AAAI Press.

Grayson, D. R. and Stillman, M. E. (1992). Macaulay2, a software system for research in algebraic geometry. `http://www.math.uiuc.edu/Macaulay2/`.

Greuel, G. and Pfister, G. (2008). *A Singular Introduction to Commutative Algebra*. Springer, Berlin, extended edition. With contributions by Olaf Bachmann, Christoph Lossen and Hans Schönemann.

Gulian, M., Frankel, A., and Swiler, L. (2020). Gaussian process regression constrained by boundary value problems. *arXiv preprint arXiv:2012.11857*.

Herlands, W., Wilson, A., Nickisch, H., Flaxman, S., Neill, D., Van Panhuis, W., and Xing, E. (2016). Scalable gaussian processes for characterizing multidimensional change surfaces. In *Artificial Intelligence and Statistics*, pages 1013–1021.

Honkela, A., Peltonen, J., Topa, H., Charapitsa, I., Matarese, F., Grote, K., Stunnenberg, H., Reid, G., Lawrence, N., and Rattray, M. (2015). Genome-wide Modeling of Transcription Kinetics Reveals Patterns of RNA Production Delays. *Proceedings of the National Academy of Sciences*, 112(42):13115–13120.

Jidling, C., Hendriks, J., Wahlström, N., Gregg, A., Schön, T. B., Wensrich, C., and Wills, A. (2018). Probabilistic Modelling and Reconstruction of Strain. *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms*, 436:141–155.

Jidling, C., Wahlström, N., Wills, A., and Schön, T. B. (2017). Linearly Constrained Gaussian Processes. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, pages 1215–1224. Curran Associates, Inc.

John, D., Heuveline, V., and Schober, M. (2019). GOODE: A Gaussian Off-The-Shelf Ordinary Differential Equation Solver. In Chaudhuri, K. and Salakhutdinov, R., editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 3152–3162, Long Beach, California, USA. PMLR.

Kocijan, J., Murray-Smith, R., Rasmussen, C. E., and Girard, A. (2004). Gaussian process model based predictive control. In *Proceedings of the 2004 American control conference*, volume 3, pages 2214–2219. IEEE.

Lange-Hegermann, M. (2018). Algorithmic Linearly Constrained Gaussian Processes. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., editors, *Advances in Neural Information Processing Systems 31*, pages 2137–2148. Curran Associates, Inc.

Levandovskyy, V. (2005). *Non-commutative Computer Algebra for polynomial algebras: Gröbner bases, applications and implementation*. PhD thesis, University of Kaiserslautern.

Levandovskyy, V. and Schönemann, H. (2003). PLURAL—a Computer Algebra System for Noncommutative Polynomial Algebras. In *Proceedings of the 2003 International Symposium on Symbolic and Algebraic Computation*, pages 176–183 (electronic). ACM.

Lima, G. S., Bessa, W. M., and Trimpe, S. (2018). Depth Control of Underwater Robots using Sliding Modes and Gaussian Process Regression. In *Proceeding of the 15th Latin American Robotics Symposium*, João Pessoa, Brazil.

Lloyd, J. R., Duvenaud, D., Grosse, R., Tenenbaum, J., and Ghahramani, Z. (2014). Automatic Construction and Natural-language Description of Nonparametric Regression Models. In *Twenty-eighth AAAI conference on artificial intelligence*.

Macêdo, I. and Castro, R. (2008). Learning Divergence-free and Curl-free Vector Fields with Matrix-valued Kernels. *Instituto Nacional de Matematica Pura e Aplicada, Brasil, Tech. Rep.*

Mayr, E. (1989). Membership in Polynomial Ideals over $\mathbb{Q}$ is Exponential Space Complete. In *STACS 89 (Paderborn, 1989)*, volume 349 of *Lecture Notes in Comput. Sci.*, pages 400–406. Springer, Berlin.

Mayr, E. W. and Meyer, A. R. (1982). The Complexity of the Word Problems for Commutative Semigroups and Polynomial Ideals. *Advances in mathematics*, 46(3):305–329.

Nguyen, N. C. and Peraire, J. (2015). Gaussian functional regression for linear partial differential equations. *Computer Methods in Applied Mechanics and Engineering*, 287:69–89.

Nguyen, N. C. and Peraire, J. (2016). Gaussian functional regression for output prediction: Model assimilation and experimental design. *Journal of Computational Physics*, 309:52–68.

Oberst, U. (1990). Multidimensional Constant Linear Systems. *Acta Appl. Math.*, 20(1-2):1–175.

Osborne, M. A., Garnett, R., and Roberts, S. J. (2009). Gaussian Processes for Global Optimization. In *3rd international conference on learning and intelligent optimization (LION3)*, pages 1–15.

Osswald, H. (2012). *Malliavin Calculus for Lévy Processes and Infinite-Dimensional Brownian Motion*.

Cambridge Tracts in Mathematics. Cambridge University Press.

Papoulis, A. and Pillai, S. U. (2002). *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill Higher Education, 4 edition.

Quadrat, A. (2010). Systèmes et Structures – Une approche de la théorie mathématique des systèmes par l'analyse algébrique constructive. Habilitation thesis.

Quadrat, A. (2013). Grade Filtration of Linear Functional Systems. *Acta Appl. Math.*, 127:27–86.

Raissi, M. (2018). Deep hidden physics models: Deep learning of nonlinear partial differential equations. *Journal of Machine Learning Research*, 19(25):1–24.

Raissi, M. and Karniadakis, G. E. (2017). Machine learning of linear differential equations using Gaussian processes.

Raissi, M. and Karniadakis, G. E. (2018). Hidden physics models: Machine learning of nonlinear partial differential equations. *Journal of Computational Physics*, 357:125–141.

Raissi, M., Perdikaris, P., and Karniadakis, G. E. (2017). Machine learning of linear differential equations using gaussian processes. *Journal of Computational Physics*, 348:683–693.

Raissi, M., Perdikaris, P., and Karniadakis, G. E. (2018). Numerical gaussian processes for time-dependent and nonlinear partial differential equations.

Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press.

Robertz, D. (2003-2008). *JanetOre: A Maple Package to Compute a Janet Basis for Modules over Ore Algebras*.

Robertz, D. (2006). *Formal Computational Methods for Control Theory*. PhD thesis, RWTH Aachen.

Robertz, D. (2015). Recent progress in an algebraic analysis approach to linear systems. *Multidimensional Syst. Signal Process.*, 26(2):349–388.

Särkkä, S. (2011). Linear Operators and Stochastic Partial Differential Equations in Gaussian Process Regression. In *International Conference on Artificial Neural Networks*, pages 151–158. Springer.

Särkkä, S. and Solin, A. (2019). *Applied stochastic differential equations*, volume 10. Cambridge University Press.

Scheuerer, M. and Schlather, M. (2012). Covariance Models for Divergence-free and Curl-free Random Vector Fields. *Stochastic Models*, 28(3):433–451.

Seiler, W. M. and Zerz, E. (2010). The Inverse Syzygy Problem in Algebraic Systems Theory. *PAMM*, 10(1):633–634.

Solin, A. and Kok, M. (2019). Know your boundaries: Constraining gaussian processes by variational harmonic features. *arXiv preprint arXiv:1904.05207*.

Solin, A., Kok, M., Wahlström, N., Schön, T. B., and Särkkä, S. (2018). Modeling and Interpolation of the Ambient Magnetic Field by Gaussian Processes. *IEEE Transactions on Robotics*, 34(4):1112–1127.

Song, Y., Zhu, J., and Ren, Y. (2016). Kernel bayesian inference with posterior regularization. In Lee, D. D., Sugiyama, M., Luxburg, U. V., Guyon, I., and Garnett, R., editors, *Advances in Neural Information Processing Systems 29*, pages 4763–4771. Curran Associates, Inc.

Sturmfels, B. (2005). What is... a Gröbner Basis? *Notices of the AMS*, 52(10):2–3.

Tan, M. H. Y. (2018). Gaussian process modeling with boundary information. *Statistica Sinica*, pages 621–648.

Thewes, S., Lange-Hegermann, M., Reuber, C., and Beck, R. (2015). Advanced Gaussian Process Modeling Techniques. In *Design of Experiments (DoE) in Powertrain Development*. Expert.

Thornton, C., Hutter, F., Hoos, H. H., and Leyton-Brown, K. (2013). Auto-WEKA: Combined Selection and Hyperparameter Optimization of Classification Algorithms. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD, pages 847–855. ACM.

Treves, F. (1967). *Topological Vector Spaces, Distributions and Kernels*. Dover books on mathematics. Academic Press.

Wahlström, N., Kok, M., Schön, T. B., and Gustafsson, F. (2013). Modeling Magnetic Fields using Gaussian Processes. In *in Proceedings of the 38th International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*.

Wu, J., Poloczek, M., Wilson, A. G., and Frazier, P. (2017). Bayesian Optimization with Gradients. In *Advances in Neural Information Processing Systems*, pages 5267–5278.

Yang, X., Tartakovsky, G., and Tartakovsky, A. (2018). Physics-informed kriging: A physics-informed gaussian process regression method for data-model convergence. *arXiv preprint arXiv:1809.03461*.

Yuan, Y., Yang, X. T., Zhang, Z., and Zhe, S. (2020). Macroscopic traffic flow modeling with physics regularized gaussian process: A new insight

into machine learning applications. *arXiv preprint arXiv:2002.02374*.

Zapała, A. M. (2002). Construction and Basic Properties of Gaussian Measures on Fréchet Spaces. *Stochastic analysis and applications*, 20(2):445–470.

Zerz, E. (2000). *Topics in Multidimensional Linear Systems Theory*, volume 256 of *Lecture Notes in Control and Information Sciences*. London.

Zerz, E., Seiler, W. M., and Hausdorf, M. (2010). On the Inverse Syzygy Problem. *Communications in Algebra*, 38(6):2037–2047.