Probabilistic Models of Motor Production

Dmytro Velychko

from Bershad, Ukraine

A dissertation submitted in partial fulfillment

of the requirements for the degree of

Doctor of natural sciences (Dr. rer. nat.)

of

Philipps University of Marburg.

Department of Psychology

Philipps University of Marburg

April 24, 2020

Fachbereich Psychologie
der Philipps-Universität Marburg (Hochshulkennzifer 1080)
als Dissertation am _____ angenommen
Erstgutauchter: Prof. Dr. Dominik Endres, Philipps-Universität Marburg
Zweigutauchter: Dr. Gunnar Blohm, Queen's University, Kingston, Canada
Tag der mündlichen Prüfung: 17 April 2020

Erklërung / Statement

Hiermit erkläre ich, dass ich meine Dissertation

"Probabilistic Models of Motor Production"

selbständig und ohne unerlaubte Hilfe angefertigt habe und mich dabei keiner anderen als der von mir ausdrücklich bezeichneten Quellen und Hilfen bedient habe. Die Dissertation wurde in der jetzigen oder einer ähnlichen Form noch bei keiner anderen Hochschule eingereicht und hat noch keinen sonstigen Prüfungszwecken gedient.

I, Dmytro Velychko, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the work. I confirm that this thesis has not been submitted to any other university or institute to obtain a degree.

Marburg, April 24, 2020

Dmytro Velychko

Acknowledgements

A curious mind is always questioning the Nature. Indeed, a child and a scientist are alike in their search and exploration of the world, asking ”why?” and ”how?”, learning from their experience or experiments. Discovering new is a challenge I could not resist, that is why I eventually ended up doing this PhD.

I would like to thank everyone who supported me throughout these years. My advisor (Doktorvater), Prof. Dominik Endres, whom I had countless discussions with. Prof. Endres must be the one who infected my brain with Bayesianism, the burden I will have to carry from now on. Being able to work with a bright person who is also willing and able to explain and share their knowledge is a real luck. And of cause special thanks to my overseas advisor, Dr. Gunnar Blohm, for his home brewed beer!

A seemingly useless but very calming hobby helped me to smooth out the roller-coaster of my PhD - learning Japanese language. I have to thank Ms. Midori Satsutani for the classes at the VHS, and my numerous acquaintances in Japan. A research visit the labs of Prof. Shigeru Shinomoto and Dr. Hideaki Shimazaki at Kyoto University was very intellectually stimulating. Of cause, being able to communicate with local people helped me a lot! I must thank DAAD for the research travel grant, I believe the submodularity in the neural code will explain the interactions dynamics of neurons beyond the simple autoencoding scheme.

It may not be a surprise, but most ideas appeared in my mind when I was not at the lab sitting in front of the screen, but simply aimlessly walking home late night or taking a shower. It seems like the very act of physical moving stimulates the brain and creative thoughts. Thus, I would like to thank my fried, Natalia Dutka for the walks we took in Marburg. Long distance running helped me to stay in a good mental and physical shape.

Finally, a lot thanks to Benjamin Knopp, Johannes Dreibrodt, Olaf Haag, and others who were involved in working with the motion capture data.

Contents

Chapter 1

Overview

N. Bernstein definierte die Fähigkeit des zentralen neuronalen Systems (ZNS) viele Freiheitsgrade eines physischen Körpers mit all seiner Redundanz und Flexibilität zu kontrollieren, als das Hauptproblem der Motorsteuerung [5]. Er wies darauf hin, dass künstliche Mechanismen normalerweise einen, manchmal zwei Freiheitsgrade (DOF) haben; Wenn die Anzahl der DOF weiter zunimmt, wird es unerschwinglich, sie zu kontrollieren. Das Gehirn scheint jedoch eine solche Kontrolle mühelos durchzuführen. Er schlug vor, wie das Gehirn damit umgehen kann: Wenn eine motorische Fähigkeit erworben wird, schränkt das Gehirn die Freiheitsgrade künstlich ein und lässt nur ein oder zwei zu. Mit zunehmendem Schwierigkeitsgrad "befreit" das Gehirn allmählich den zuvor festgelegten DOF und wendet bei Bedarf die Steuerung in Richtungen an, die korrigiert werden müssen, um schlielich das Steuerungsschema zu erreichen, bei dem alle DOF "frei" sind. Dieser Ansatz zur Reduzierung der Dimension der Motorsteuerung ist auch heute noch relevant.

Eine der möglichen Lösungen für das Bernstetin-Problem ist die Hypothese von *motor primitives* (MPs) - kleinen Bausteinen, die komplexe Bewegungen darstellen und das motorische Lernen und die Erledigung von Aufgaben erleichtern. Genau wie im visuellen System kann es von Vorteil sein, eine homogene hierarchische Architektur zu haben, die aus ähnlichen Rechenelementen aufgebaut ist [63].

Bei der Untersuchung eines so komplizierten Objekts wie des Gehirns ist es wichtig zu definieren, auf welcher Detailebene gearbeitet wird und welche Fragen beantwortet werden sollen. David Marr [49] schlug drei Analyseebenen vor:

1. computational, analysieren *welches Problem* das System löst; 2. algorithmic, Abfrage *welche Darstellung* das System verwendet und *welche Berechnungen* es durchführt; 3. implementational, herrausfinden von *wie solche Berechnungen von Neuronen im Gehirn durchgeführt werden*. In dieser Arbeit bleiben wir auf den ersten beiden Ebenen und suchen nach der grundlegenden Darstellung der Motorleistung.

In dieser Arbeit stellen wir ein neues Modell von motorischen Primitiven vor, das mehrere interagierende latente dynamische Systeme umfasst und eine vollständige Bayessche Behandlung erlaubt. Das Modellieren im Bayessche Rahmen muss meiner Meinung nach der neue Standard für das Testen von Hypothesen in den Neurowissenschaften werden. Nur der Bayessche Rahmen gibt uns Garantien, wenn es um die unvermeidliche Fülle von latenten Variablen und Unsicherheiten geht [P1, P2, P3].

Die spezielle Art der Kopplung von dynamischen Systemen, die wir in [P1], basierend auf dem Produkt von Experten [33], vorgeschlagen haben, hat viele natürliche Interpretationen im Bayessche Rahmen. Wenn die dynamischen Systeme parallel laufen, ergibt sich eine Bayessche Cue-Integration. Wenn sie aufgrund der seriellen Kopplung hierarchisch organisiert sind, erhalten wir hierarchische Prioritäten über die Dynamik. Wenn eines der dynamischen Systeme den sensorischen Zustand repräsentiert, kommen wir zu den sensor-motorischen Primitiven. Die kompakte Darstellung, die sich aus der variationellen Behandlung ergibt, ermöglicht das Lernen einer Bibliothek motorischer Primitiven. Separat gelernt, kann die kombinierte Bewegung als Matrix von Kopplungswerten dargestellt werden.

Wir haben eine Reihe von Experimenten durchgeführt, um verschiedene Modelle von Motorprimitiven zu vergleichen [P2, P3]. In einer Reihe von 2-Alternative-Forced-Choice-Experimenten (2AFC) unterschieden die Teilnehmer natürliche und synthetisierte Bewegungen und führten so einen graphischen Turing-Test durch. Sofern verfügbar, sagte der Bayessche Modellwert die Natürlichkeit der wahrgenommenen Bewegungen voraus. Für einfache Bewegungen wie das Gehen

zeigen der Bayessche Modellvergleich und psychophysische Tests, dass ein dynamisches System ausreicht, um die Daten zu beschreiben. Bei komplexeren Bewegungen wie Gehen und Winken kann die Bewegung besser als ein Zusammenschluss mehrerer gekoppelter dynamischer Systeme dargestellt werden. Wir haben auch experimentell bestätigt, dass die Bayessche Behandlung des Modelllernens an Bewegungsdaten der einfachen Punktschätzung latenter Parameter überlegen ist. Experimente mit nichtperiodischen Bewegungen zeigen, dass sie trotz hoher kinematischer Komplexität nicht von einer komplexeren latenten Dynamik profitieren.

Mit einem vollständig bayesianischen Modell könnten wir den Einfluss der Bewegungsdynamik und Pose auf die Wahrnehmung von Natürlichkeit *quantitativ* entflechten. Wir haben bestätigt, dass eine umfassende und korrekte Dynamik wichtiger ist als die kinematische Darstellung.

Es gibt zahlreiche weitere Forschungsrichtungen. In den Modellen, die wir für mehrere Teile entwickelt haben, waren die kinematischen Teile völlig unabhängig, obwohl die latente Dynamik auf einer Reihe von interagierenden Systemen faktorisiert wurde. Die Wechselwirkung zwischen den kinematischen Teilen konnte also nur durch die latenten dynamischen Wechselwirkungen vermittelt werden. Ein flexibleres Modell würde eine Interaktion auch auf kinematischer Ebene ermöglichen.

Ein weiteres wichtiges Problem betrifft die *Darstellung der Zeit* in Markov-Ketten. Diskrete Zeit Markov-Ketten sind eine Annäherung an die kontinuierliche Dynamik. Da angenommen wird, dass der Zeitschritt festgelegt ist, stehen wir vor dem Problem der Zeitschrittauswahl. Zeit ist auch in Markov-Ketten kein expliziter Parameter. Dies verbietet auch eine explizite Optimierung der Zeit als Parameter und eine Folgerung (Inferenz) darüber. Beispielsweise werden bei einer optimalen Steuerung die Randbedingungen normalerweise zu genauen Zeitpunkten festgelegt, was kein ökologisches Szenario ist, bei dem die Zeit normalerweise ein Parameter der Optimierung ist. Wenn Sie die Zeit zu einem expliziten Parameter in der Dynamik machen, kann dies möglicherweise Abhilfe schaffen.

In den nächsten Kapiteln geben wir einen kurzen überblick über die Mo-

torsteuerung und die Motivation hinter der Verwendung von Motorprimitiven für dieses Problem: einen kurzen überblick über die Bayessche Statistik und warum dies wichtig ist, gefolgt von Kapitel 3 über Motorprimitive und Motorsteuerung im Gehirn. Kapitel 4 befasst sich mit der motorischen Steuerung durch das Gehirn, und Kapitel 5 behandelt den Gauschen Prozess und seine Verbindung mit künstlichen neuronalen Netzen. Diese Kapitel sollen dem Leser helfen, die Zusammenhänge zwischen Bayessche Statistik, Gau-Prozess, neuronalen Netzen und Motorsteuerung zu verstehen.

N. Bernstein defined the ability of the central neural system (CNS) to control many degrees of freedom of a physical body with all its redundancy and flexibility as the main problem in motor control [5]. He pointed at that man-made mechanisms usually have one, sometimes two degrees of freedom (DOF); when the number of DOF increases further, it becomes prohibitively hard to control them. The brain, however, seems to perform such control effortlessly. He suggested the way the brain might deal with it: when a motor skill is being acquired, the brain artificially limits the degrees of freedoms, leaving only one or two. As the skill level increases, the brain gradually ”frees” the previously fixed DOF, applying control when needed and in directions which have to be corrected, eventually arriving to the control scheme where all the DOF are ”free”. This approach of reducing the dimensionality of motor control remains relevant even today.

One the possibles solutions of the Bernstetin's problem is the hypothesis of *motor primitives* (MPs) - small building blocks that constitute complex movements and facilitite motor learnirng and task completion. Just like in the visual system, having a homogenious hierarchical architecture built of similar computational elements may be beneficial [63].

Studying such a complicated object as brain, it is important to define at which level of details one works and which questions one aims to answer. David Marr [49] suggested three levels of analysis: 1. computational, analysing *which problem* the system solves; 2. algorithmic, questioning *which representation* the system uses and *which computations* it performs; 3. implementational, finding *how such computations are performed by neurons in the brain*. In this thesis we stay at the first two levels, seeking for the basic representation of motor output.

In this work we present a new model of motor primitives that comprises multiple interacting latent dynamical systems, and give it a full Bayesian treatment. Modelling within the Bayesian framework, in my opinion, must become the new standard in hypothesis testing in neuroscience. Only the Bayesian framework gives us guarantees when dealing with the inevitable plethora of hidden variables and uncertainty [P1, P2, P3].

The special type of coupling of dynamical systems we proposed in [P1], based on the Product of Experts [33], has many natural interpretations in the Bayesian framework. If the dynamical systems run in parallel, it yields Bayesian cue integration. If they are organized hierarchically due to serial coupling, we get hierarchical priors over the dynamics. If one of the dynamical systems represents sensory state, we arrive to the sensory-motor primitives. The compact representation that follows from the variational treatment allows learning of a motor primitives library. Learned separately, combined motion can be represented as a matrix of coupling values.

We performed a set of experiments to compare different models of motor primitives [P2, P3]. In a series of 2-alternative forced choice (2AFC) experiments participants were discriminating natural and synthesised movements, thus running a graphics Turing test. When available, Bayesian model score predicted the naturalness of the perceived movements. For simple movements, like walking, Bayesian model comparison and psychophysics tests indicate that one dynamical system is sufficient to describe the data. For more complex movements, like walking and waving, motion can be better represented as a set of coupled dynamical systems. We also experimentally confirmed that Bayesian treatment of model learning on motion data is superior to the simple point estimate of latent parameters. Experiments with non-periodic movements show that they do not benefit from more complex latent dynamics, despite having high kinematic complexity.

By having a fully Bayesian models, we could *quantitatively* disentangle the influence of motion dynamics and pose on the perception of naturalness. We confirmed that rich and correct dynamics is more important than the kinematic representation.

There are numerous further directions of research. In the models we devised, for multiple parts, even though the latent dynamics was factorized on a set of interacting systems, the kinematic parts were completely independent. Thus, interaction between the kinematic parts could be mediated only by the latent dynamics interactions. A more flexible model would allow a dense interaction on the kinematic level too.

Another important problem relates to the *representation of time* in Markov chains. Discrete time Markov chains form an approximation to continuous dynamics. As time step is assumed to be fixed, we face with the problem of time step selection. Time is also not a explicit parameter in Markov chains. This also prohibits explicit optimization of time as parameter and reasoning (inference) about it. For example, in optimal control boundary conditions are usually set at exact time points, which is not an ecological scenario, where time is usually a parameter of optimization. Making time an explicit parameter in dynamics may alleviate this.

In the next chapters we give a brief overview of motor control and motivation behind the use of motor primitives for this problem: a brief overview of Bayesian statistics and why it is important in, followed by a chapter 3 on motor primitives and motor control in the brain, extended further in chapter 4 by theories applicable to tackle the question of motor control by the brain, and finalizing by a chapter 5 on Gaussian Process and its connection to artificial neural networks. These chapters are intended to help the reader to understand the connections between Bayesian statistics, Gaussian Process, neural networks, and motor control.

Publications and contributions

[P1] Dmytro Velychko, Benjamin Knopp, and Dominik Endres. “The Variational Coupled Gaussian Process Dynamical Model”. In: *Artificial Neural Networks and Machine Learning - ICANN 2017 - 26th International Conference on Artificial Neural Networks, Alghero, Italy, September 11-14, 2017, Proceedings, Part I*. 2017, pp. 291–299. Velychko: derived the model, implemented it in python, collected and processed the motion capture data, analysed the psychophysical data; contribution 60%. Knopp: run the psychophysics experiment, analysed the psychophysical data. Endres: supervised. All authors: wrote paper.

[P2] Dmytro Velychko, Benjamin Knopp, and Dominik Endres. “Making the Coupled Gaussian Process Dynamical Model Modular and Scalable with Variational Approximations”. In: *Entropy* 20.10 (2018), p. 724. Velychko: derived the model, implemented it in python, collected and processed the motion capture data, analysed the psychophysical data; contribution 60%. Knopp: run the psychophysics experiment, analysed the psychophysical data. Endres: supervised. All authors: wrote paper.

[P3] Benjamin Knopp et al. “Predicting Perceived Naturalness of Human Animations Based on Generative Movement Primitive Models”. en. In: *ACM Transactions on Applied Perception* 16.3 (2019), pp. 1–18. Velychko: collected and processed the motion capture data, wrote the model training code; contribution 40%. Knopp and Dreibrodt: run the psychophysics experiment, analysed the psychophysical data. Endres: supervised. Knopp, Dreibrodt, Endres: wrote paper.

Chapter 2

Bayesian Statistics

"Given for one instant an intelligence which could comprehend all the forces by which nature is animated and the respective situation of the beings who compose it—an intelligence sufficiently vast to submit these data to analysis—it would embrace in the same formulate the movements of the greatest bodies of the universe and those of the lightest atom; for it, nothing would be uncertain and the future, as the past, would be present in its eyes."

— P.-S. Laplace, *A philosophical Essay on Probabilities*

2.1 Reasoning about uncertainty

When dealing with any real world system, one has to account for the uncertainty either implicitly (e.g. Tikhonov regularization[82], decision boundaries in Empirical Risk Minimisation [73]) or explicitly, in Bayesian way, clearly stating the prior beliefs and accounting for the full (approximate) posterior. A rigorous motivation behind probability in reasoning about uncertainty was given by Frank Ramsey [66, 32]. According to him, a rational agent, that accepts some intuitively reasonable criteria of rationality, must follow the rules of probability when reasoning about environments that are deterministic but not fully observed or stochastic in nature.

Formally, σ-algebra is a collection of sets closed under complement and countable unions. A triplet of (Ω, F, P), where Ω is a nonempty base set of possible outcomes, F is σ-algebra over Ω - a set of events, and P is a normalized to 1

measure on (Ω, F), is a *probability space* [43, 7]. Random variable is a function of elementary outcomes [43]. From the properties of measure follows that probability of an event is the sum of probabilities of elementary outcomes it contains: $P(X) = \sum_{x_i \in X} p(x_i); x_i \in \Omega$.

Next, we may introduce the concepts of *joint* and *conditional probabilities*. Joint probability of two events A and B is equivalent to the probability of a single event $A \cap B$. *Conditional probability* $P(A|B)$ may be introduced by any of the following equivalent statements: conditioning of probability measure P on event B creates a new measure P_B

- by only considering such possible worlds (outcomes) where the event B happens;
- by setting the measure of the event complementary to B to 0. The remaining elementary outcomes in event B have to be renormalized to make P_B a valid probability measure.

From the whole set of possible outcomes Ω we consider only those which are contained in B. Thus, we limit the space of possible events to be consistent with the fixed (observed) B; probability of the event complementary to B is 0:

$$P(A|B) \coloneqq P_B(A) \tag{2.1}$$

$$P_B(B) = 1 \tag{2.2}$$

$$P_B(\neg B) = 0 \tag{2.3}$$

$$P_B(A) \propto P(A, B) \tag{2.4}$$

This changes the probability measure over F and requires additional renormalization by the support set of B:

$$P(A|B) = \frac{P(A, B)}{\sum_{b \in B} p(b)} \tag{2.5}$$

$$= \frac{P(A, B)}{P(B)} \tag{2.6}$$

This introduction of conditional probability is consistent with Kolmogorov's definition [43]. Notice that $P(A|B)$ is a function of B and a probability measure over A. From this, by simple algebraic manipulations we get the following two possible factorizations of joint probability:

$$P(A, B) = P(B|A)P(A) \tag{2.7}$$

$$= P(A|B)P(B) \tag{2.8}$$

The Bayes' formula now is just a step away:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{2.9}$$

It does not provide any additional advantage over the original concept of conditional probability and completely relies on it, but has a nice interpretation when the joint probability $P(A, B)$ is described as a Bayesian network.

A causal stochastic process with a set of hidden and observed variables can be represented as a directed acyclic graph, a Bayes network [61]. Even though the full factorization of any joint probability of the variables can be performed in many ways, the factorization that represent causal links between them has a meaningful and useful interpretation. If we consider a simple causal graph $A \rightarrow B$, by conditioning on observed B we may infer the probability distribution of unobserved causes A, if our assumptions about the process (joint distribution of A and B) are correct. Thus, the Bayes' formula equips us with a mathematically solid method of probabilistic model inversion, computing posterior beliefs conditioned on observations.

2.2 Model comparison

The Bayes' formula rescues us when we are not sure about the underlying data generating process and have multiple hypothesis. Assume two models (hypotheses) about how the data X was generated, M_a and M_b with parameters θ. Then

comparing the models boils down to inferring the posterior distribution over them:

$$p(X|M_i) = \int p(X|\theta, M_i)p(\theta|M_i)d\theta \tag{2.10}$$

$$p(M_i|X) = \frac{1}{Z_M}\int p(X|\theta, M_i)p(\theta|M_i,)p(M_i)d\theta \tag{2.11}$$

where $i \in \{a, b\}$, $Z_M = \sum_{j\in\{a,b\}} p(M_j|X)$ is a normalizing constant.

For reasonably complex datasets (e.g. containing continuous variables) we can construct an infinite space of possible models (hypotheses) explaining the data [1]. To compare two parametric models, we can compute the ratio of their probabilities, known as *Bayes factor*:

$$F_{\frac{a}{b}} = \frac{p(M_a|X)}{p(M_b|X)} = \frac{p(M_a)\int p(X|\theta, M_a)p(\theta|M_a)d\theta}{p(M_b)\int p(X|\theta, M_b)p(\theta|M_b)d\theta} \tag{2.12}$$

An attentive reader may now notice that in Bayesian framework, model comparison is just conditioning on data in a model that comprises the models to be compared linked by an additional model selection variable on top – inference over a single discrete latent variable.

2.3 Learning as belief update

In the problem of model identification one may be faced the questions of how many parameters make a good model, will it result in overfitting or underfitting the available data [8]. The Bayes' rule solves such problems. Given the model is correct and the prior distribution has a reasonable support (probability of true parameters is not zero), updating to the posterior always accounts for the data properly, even if the data size is only one sample.

Imagine we have an intelligent agent in a complex environment. Be the environment fully observed for a moment, and the agent were equipped with infinite

[1] Such model space comprises every possible generative process of the data. We may also create such model space by constructing a mapping from an infinite space of finite strings (in terms of Kolmogorov complexity theory - functions that generate the data) to the model space, thus, enumerating them. As the space of possible models in this setting is potentially infinite, we can only compare distinct models (points in the model space, or strings in the function definition space) relatively to each other.

computational resources, it would be possible for him to simulate and predict the future simply integrating its evolution function as well as any effect of his actions. If, however, such environment is intrinsically stochastic, or deterministic but not completely observable, the agent must deal with it as with a stochastic system. In such environment being rational means updating the beliefs representing the hidden variables according to the Bayes rule. This setting is similar to the one Laplace [45] used to justify proper treatment of uncertainty as unavoidable feature of human knowledge and treating it according to the rules of probability.

2.4 Variational Inference

Exact Bayesian inference requires computing the normalization constant (renormalizing the conditioned hypothesis space), which is usually an intractable problem. Numerous approximations were developed to add to its tractability [8].

The simplest one is just point estimate of the posterior after conditioning - *maximum a posteriori* (MAP). It seeks for such parameter value that maximizes the joint model probability:

$$\theta_{MAP} = \arg\max_{\theta} p(data, \theta) \tag{2.13}$$

MAP completely ignores the probability mass or other possible modes [47]. Thus, contrary to the full Bayesian update, MAP approximation may lead to rather incorrect parameter values.

A popular approximation scheme is *variational inference*, when an arbitrary parametrized distribution is fitted to the true posterior by maximizing the probability of observed data (model evidence) [8].

Consider a simple model $p(X, \theta) = p(X|\theta)p(\theta)$ where observed sensory data X is parametrised by a set of hidden variables: $\theta \rightarrow X$. Let us introduce a parametric recognition probability density $q(\theta|X)$ - a probabilistic mapping from observation X to hidden variables θ, which may represent the observer's beliefs about the environment. Recognition density has a natural interpretation in the Bayesian brain framework: it is a learned approximation of the true posterior of causes θ having

sensory input X, that can be quickly computed. For any model its evidence reads:

$$p(X) = \int p(X|\theta)p(\theta)d\theta \tag{2.14}$$

By adding mutually cancelling $\exp\log$ transformation and introducing the recognition density we get:

$$p(X) = \int p(X,\theta)d\theta = \exp\log \int q(\theta|X)\frac{p(X,\theta)}{q(\theta|X)}d\theta \tag{2.15}$$

We will use the Jensen's inequality [8] for convex functions:

$$f(\mathbb{E}(X)) \leq \mathbb{E}(f(X)) \tag{2.16}$$

In integral form it reads:

$$f\left(\int g(x)p(x)dx\right) \leq \int f(g(x))p(x)dx \tag{2.17}$$

A very handy convex function is $f(x) = -\log(x)$. It often simplifies computations when working with probabilities as $\log(a \cdot b) = \log(a) + \log(b)$.

Finally, by taking the $\log$ of both sides of Eq. 2.15 and applying the Jensen's inequality we get the model evidence lower bound (ELBO):

$$\log p(X) \geq \int q(\theta|X)\log\frac{p(X,\theta)}{q(\theta|X)}d\theta \tag{2.18}$$

$$ELBO = \int q(\theta|X)\log\frac{p(X,\theta)}{q(\theta|X)}d\theta \tag{2.19}$$

$$= \int q(\theta|X)\log p(X,\theta)d\theta - \int q(\theta|X)\log q(\theta|X)d\theta \tag{2.20}$$

Now we may optimize ELBO w.r.t. the parameters of the recognition density $q(\theta|X)$. Such optimization of ELBO can be interpreted as minimization of Gibbs free energy and forms one of the core principles of the Free Energy brain hypothesis [23].

In my work I used ELBO to find the approximate posterior and compute the

Bayes factors. It also allowed me to compare the contributions of different parts of the model to perceptual validity.

Chapter 3

Brain and Motor Control

Until now the motor cortex remains not well understood, raising numerous hypothesis about the representations and computations it performs, as neural activity correlates with separate muscle activations, synergies, complex movements, kinematics, dynamics, goal, and many other features of movements [58, 38, 25, 75, 76]. Surely, cerebellum, basal ganglia, sensory areas, frontal lobe are impossible to ignore when it comes to sensorimotor learning, precise control, planning and decision making [44]. Here we give a quick overview of some experiments that shed light on the functional role of motor areas of the brain and their ability to generate motor output.

The main function of the brain, besides regulation of homeostasis, is to process sensory information and consequently produce motor output. Brain is required for motion. Llinas [46] indicates that sea squirt in its larval form has a tiny (approximately 300 cells, 100 of which are neurons) brain-like ganglion that allows it to move through the environment, while in its adult form it attaches itself to a substrate and digests most of its brain. (It is worth mentioning that this statement is questionable, Mackie and Burighel [48] point out: ”In fact, adult ascidians have perfectly good brains, an order of magnitude larger than those of their larvae, and their behaviour is as finely adapted to sessility as that of the larvae to motility.”) It is easy to extend motor control on regulation of homeostasis too, one just has to consider physical body as a part of the environment. From this very general perspective brain is just a large function that converts sensory input into output signals, affect-

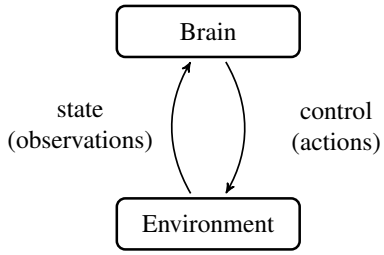


Figure 3.1: Brain-environment interaction loop. If body is considered to be a part of the controlled environment, homeostasis control follows the same principles as motor control.

ing the exogenous environment via activation of muscles or modulating the release of hormones for the endogenous part of the environment (Fig. 3.1). The free energy principle [23] makes no distinction between such environments, thus unifying these two kinds of actions.

3.1 Motor programs

The lowest level of motor control is performed by spinal cord. From the studies of locomotion in decerebrated cats [29] we may conclude that spinal cord alone is able to generate different walking patterns. Different periodic patterns of activations can be observed even in an in-vitro preparation of brainstem and spinal cord [77]. Even deafferenated mesencephalic cat can produce a locomotor pattern [31, 30]. This indicates that even in mammals the very simple patterns of behaviour and locomotion can be driven by relatively simple neural circuits developed in the neural system in the early stages of development. Such simple neural circuits are called *central pattern generators* (CPGs) and are considered to be the core elements of locomotion generation mechanism in insects and animals [72].

As for the primary motor cortex (M1), for a long time it had been viewed as a fine controller, able to activate a single muscle. This idea comes from short (10-50 ms) microstimulation studies, when each such electric stimulation of different regions of M1 could produce a reproducible twitch of different muscles (see e.g. [19] for the first experiments on mapping of monkey brain). This led to a rather simple cortical map of the whole body onto M1.

Neural activity in motor and premotor cortex areas is, however, rather complex. It can be correlated with many features of motor production, like direction, speed,

force (see e.g. [24] for the population coding of direction in M1). Recently it was demonstrated that a much longer, with behaviorally relevant duration of 500 ms, local stimulation of the primary and premotor cortex in monkeys evokes complex and semantically meaningful motions [28, 27]. Such maps represent the complex repertoire of movements specific to a particular monkey [26]. Perhaps the most complex movement that was initiated by such type of stimulation is the one resembling a defensive action [28, 13], when particular regions in polysensory zone (PZ) were excited. It recruits almost the whole body: "Stimulation of these sites evoked a constellation of movements including blinking, squinting, flattening the ear against the side of the head, elevating the upper lip, shifting the head away from the sensory receptive fields, shrugging the shoulder, and rapidly lifting the hand into the space near the side of the head as if to block an impending impact" [13]. The monkey did not avoid an obstacle placed on the limb path, bumping into it. Neuronal activity in that area was highly correlated with defence behaviour triggered by an air puff to the face. Such complex movements could be induced also in monkeys anaesthetised with barbiturates. Thus, Graziano argued that it was not a reaction to a fictive sensory input. It is hard to say, however, what kind of sensation monkeys had during the stimulation.

These motor maps can change and evolve as animals acquire skills [51] or get trained to perform specific tasks. Experiments with rats [65] demonstrated that when a region corresponding to some stereotypical movement (reaching) is lesioned, the rat lost the ability to perform that movement. After the recovery and re-training a new region could be found near the lesion site that would evoke the same reaching movement if stimulated.

In another study on monkeys [59] the complexity and statistical properties of the microstimulation induced movements were analysed. The recovered muscle synergies and their statistics from intracortical stimulation matched the ones of naturally performed reaching and grasping movements. This supports the hypothesis that motor control learning is driven by the natural environment statistics, so successfully applied to the analysis of visual system receptive fields formation [57].

The authors conclude that rich voluntary movements originated in cortex are hypothetically performed by a complex temporal activation of the underlying muscle synergies circuits.

While the CPGs and muscle synergies simplify the problem of locomotion generation and simple reaching-and-grasping, fine motor control and precise object manipulation seem to rely more on cotrical control. Neural pathway tracing with rabies virus in monkeys [68] reveal two types of connections of M1 to motor neurons. The "old M1" corticospinal neurons innervate motor interneurons, which are known to be recruited in the reflexes, CPGs, and muscle synergies [72]. The "new M1" cortico-motoneuronal cells have direct connection to motor neurons, thus allowing the precise cortical control on single muscle level. The proportion of such direct connections correlates with the dexterity of different species.

Observations and motor control experiments with humans lacking proprioceptive signals (damaged afferent neural pathways) [69, 50] suggest that rich motor production is possible without any sensory feedback. "The patient could perform a surprising range of other rather less complex motor tasks even with his eyes closed. Thus individual digits could be activated independently without practice and he could even touch his thumb with each finger in turn using either hand" [69]. He could even draw complex patterns in the air with fingers with his eyes closed, and drive a manual shift car in complete darkness. Surprisingly, precision of fast movements was not different from health subjects, even without visual feedback. Only in tasks that required precise feedback, like slow movements, stabilization (constant force, constant position, walking, etc.), fine force control, performance was degraded. The longer the task lasted, the worse was the performance due to the accumulation of errors. The authors conclude that the timing, sequence, and patterns of muscle activations, which they call "motor programmes", could be executed without any external feedback when somatosensory feedback was crucial for tasks that require error correction.

Experiments with healthy participants [15] demonstrated that a very small number of components of muscle activation patterns is enough to perform a large

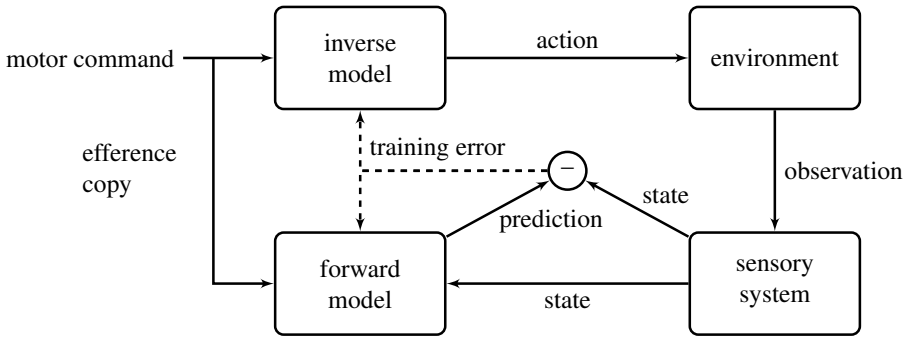


Figure 3.2: Internal models in the brain. Forward model receives a copy of the motor command, old state estimation, and produces the next sensory estimation. The difference between the prediction and the actual state estimation drives the learning of both the forward and inverse models. (Adapted from [52, 40])

set of fast-reaching movements. Such temporal activation patterns, or synergies, form a low-dimensional manifold and greatly simplify planning and control. 4-5 of of such synergies were enough to explain 73-82% of the data variability.

3.2 Sensory Feedback

We have seen that motor production is possible without any sensory input. For precise motions, sensory data provides the necessary information to adjust the motor commands. Motor control and learning, thus, are the main research directions in the field of biological motor production.

One can argue that the role of the motor system feedback is to deal with sensory and motor constrains and imperfections, to compensate for the stochasticity of the environment, muscles, and the brain itself. Muscle response properties are particularly volatile. Noise in motor system scales proportionally to the signal amplitude. Temporal delays, which are inevitable in neural information processing systems, further reduce the controllable subspace.

Largely inspired by engineering approaches, mathematical optimal control theory, and Bayesian multisensory integration, the biological motor system is viewed as a chain of transformations of concrete motor commands into muscle activations and an error compensating feedback controller.

For robust feedback control in an uncertain environment one has to have an approximate model of that environment, especially when the controller has delays

in information processing, like in every neural system. This resulted in the hypothesis about *forward and inverse models* of the controlled environment [90, 40] (Fig 3.2). Motor commands are neural activations that represent motion in some abstract space (of sensory expectations). Inverse model maps a motor command onto muscle activations, while forward model predicts the sensory outcomes. Prediction error drives the learning and adaptation of the models. Physiological recordings from cerebellum, lesion studies, and clinical observations give a lot of evidence that the brain may implement the forward and inverse models for short time scales [89, 17, 81]. For planning and longer time scales, more cortical areas are recruited, like posterior parietal cortex for visually guided movements [9], and frontal lobe for decision making [12], essentially rendering the whole brain a feedback controller.

In the next chapter we will discuss some approaches aiming to explain computations required for brain to be a motor controller, residing on the computational level of the Marr's hierarchy.

In my work CPGs and muscle synergies naturally relate to the latent dynamical systems and latent to observed mappings. Hierarchical coupling may represent the top-down activation of the CPGs. Sensory coupling relates to the feedback control.

Chapter 4

Approaches to Motor Production

While the problem of perception, at least in principle, can be solved by employing Bayesian inference [42], perception alone is not sufficient to make an agent *act*. Optimal control and reinforcement learning explain actions as optimization of some reward (or cost) function. In free energy framework [23] actions affect the environment to change it into accordance to internal beliefs. Bayesian motor control explains *how* to reduce the motor control problem to a probabilistic inference. Here we overview these theories of motor production.

4.1 Optimal Control. Continuous time

Some of the observed phenomena of motor production, like optimality, variance of trajectories, can be explained if approached from the optimal control and decision theory perspective [84, 83]. Body and environment dynamics can be described by a system of stochastic differential equations with certain constraints:

$$dx = f(x, u)dt + g(x, u)dW \tag{4.1}$$

where x is the full system state vector, u is the control signal emitted by the neural system, and W is Wiener process (white noise) [56]. Matrix-valued functions f and g are called *drift* and *diffusion* coefficients correspondingly and represent deterministic and stochastic components of the dynamics, like muscle forces and body dynamics under neural control signal $f(x, u)dt$ and control multiplicative

noise $g(x, u)$. By providing a cost functional

$$C(x(\cdot), u(\cdot)) = \int_0^T l(x(t), u(t), t)dt \tag{4.2}$$

which gives the expectation of the integral w.r.t. of all the possible stochastic trajectories starting from x_0, we seek for the optimal control function $u^*(\cdot)$ that minimizes C:

$$u^* = \arg\min_u \mathbb{E}_{x0} \left[C(x(\cdot), u(\cdot))\right] \tag{4.3}$$

In general case, due to the nonlinearity and high dimensionality this problem is unrealistic to solve exactly, but assuming some additional constraints and structure on the dynamics and cost functions, exact solution for the classical Linear-Quadratic-Gaussian (LQG) control problem exist. For non-linear dynamics an interesting and promising Path Integral Control [39] approach reduces the problem to a path integral ([37]) which can be approximately solved by sampling or free energy minimization, effectively creating a link to the Bayesian methods.

4.2 Optimal Control. Discrete time

Dealing with continuous time makes us work with differential equations. By switching to discrete time, we can employ many powerful numerical optimization methods and thus hope to solve more complex problems. Here we just give a formulation starting from *decision process* in discrete time, and describe an important statement on which the solution hinges - the Bellman's optimality principle.

For a fully observed environment and discrete time, Markov Decision Process (MDP) [6, 78] can be formulated as follows : given a probabilistic state transitioning matrix $p(s_{t+1}|s_t, a_t)$ and stationary immediate reward probabilities $r(s_{t+1}, s_t, a_t) \sim p(r|s_{t+1}, s_t, a_t)$, produce actions to maximize the expected cumulative reward. Time horizon can be set fixed ($t_{max} = T$), or infinite. In the latter case we are interested in the sum of exponentially discounted rewards. If the environment has hidden variables the agent can not observe, we say that the decision process is partially

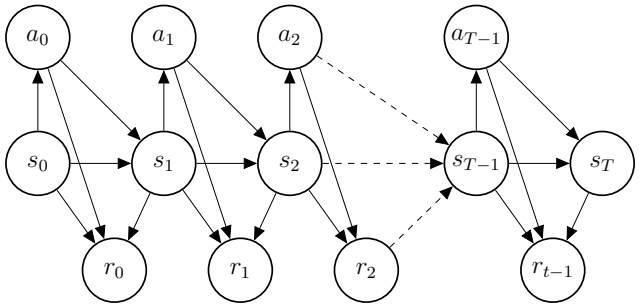


Figure 4.1: Markov decision process. At every time step t the agent performs an action a_t, causing the environment to change its state to s_{t+1}, and receive reward r_t. In graphical models circles represent random variables, and arrows indicate the direction of statistical or causal dependencies. [62].

observable (POMDP).

Generally, solving an MDP results in a policy - a probabilistic function that tells which action to take given the observed state, $a \sim \pi(a|s)$.

Let's consider a fixed time horizon MDP with finite state space. Expected immediate reward following some policy π is a reward expectation w.r.t. the next state, and reads:

$$R_\pi(s_t) = \mathbb{E}_{s_{t+1},a_t}[r(s_{t+1}, s_t, a_t)\pi(a_t|s_t)] \tag{4.4}$$

Next, we introduce *value function* for each state as the expected reward following some policy π:

$$V_\pi(t, s) = \mathbb{E}_{s_{t+1}\ldots s_T|\pi}\left[\sum_{i=t}^{T} R_\pi(s_i) \mid s_t = s\right] \tag{4.5}$$

where the expectation is taken w.r.t. all possible trajectories under the policy π.

Notice that isolating out one decision step, the value function reveals the following recursive property:

$$V_\pi(t, s) = R_\pi(s_t) + \mathbb{E}_{s_{t+1}|s_t,\pi}\left[V_\pi(t+1, s_{t+1})\right] \tag{4.6}$$

$$= R_\pi(s_t) + \sum_{a_t} \pi(a_t|s_t) \sum_{s_{t+1}} [p(s_{t+1}|s_t, a_t)V_\pi(t+1, s_{t+1})] \tag{4.7}$$

This results in the following recursive equations, which formulate the Bell-

man's optimality principle [4]:

$$V^*(t,s) = \max_a \left[r(s_t,a) + \mathbb{E}_{s_{t+1}|s,a} \left[V^*(t+1, s_{t+1}) \right] \right] \tag{4.8}$$

$$\pi^*(a|s) = \arg\max_a \left[r(s_t,a) + \mathbb{E}_{s_{t+1}|s,a} \left[V^*(t+1, s_{t+1}) \right] \right] \tag{4.9}$$

$$r(s_t,a) = \mathbb{E}_{s_t+1} \left[r(s_{t+1}, s_t, a) \right] \tag{4.10}$$

These equations, which iterate backwards in time constructing the full value function and the optimal policy, are the core of dynamic programming. The complexity of dynamic programming resides in the size of the value function, which is $\prod_{i=1...N} D_i$, where D_i is the range of $i-$th state variable. Due to the exponential increase in the size of the optimal value function and policy their applicability is limited only to problems with small state space. For continuous domains one can use fine discretization, but such approach works only for very simple state spaces. When the reward function is unknown and has to be learned at the same time, the problem is formulated as Reinforcement Learning (RL) [78].

Despite the complexity of the problem, by making a reasonable assumption that in real world the value function complexity has some structure and can be approximated well by a complex parametrized functions (e.g. deep neural networks), iterative optimization of $V(s)$ was demonstrated to reach an impressive performance [53]. On the other side, simple linear or RBF approximations of the policy function is sufficient to solve many optimal control problems faster and with better cost, indicating that end-to-end deep RL has an advantage mostly because of the elaborate sensory system they provide [64].

Optimal control in redundant setting, when the number of controllable dimensions is larger than required to perform the task, also gives rise to correlations between muscle activations, and synergies [83]. The so-called *controlled manifold*, the part of the state space which is actually affected by the control signal, is relatively low-dimensional; when the noise model is control-multiplicative, as is it often the case in biological systems [36], any control signal adds more noise to the

system, thus control in dimensions which do not affect task performance is harmful and not needed. It is known as *minimum intervention principle* [83]. This had been suggested as the solution for the dimensionality problem for biological motor production.

So far the best performing optimal motor control methods rely on crafting the cost functions, discrete time, iterative optimization, differentiable environments, or even manually engineered and tuned solutions.

Learning from rewards in natural environment usually sets another problem - as the rewards are usually sparse and distant in time from the actions that led to them, identifying the correct action to reinforce becomes a non-trivial task. This problem is knows as *credit assignment*,

Reinforcement learning paradigm had been successfully applied to explain some aspects of learning in the brain, conditioning, dopaminergic system. Probably, correlation of dopamine dynamics with some of learning signals in temporal difference algorithm as reward system representing reward prediction error [54, 74],

As for biological plausibility of optimal control, I would like to indicate some of the important incoherences:

- Optimal control requires the value function, defined explicitly, or learned, as in the Reinforcement Learning paradigm.

- Even though movements can be expressed as optimal solutions w.r.t. some cost, it may not be a consequence of optimizing that function.

- Optimal control is invariant to the learning procedure, it explains only the end-result. From the Marr's perspective [49], optimal control resides on the top (computational) level.

A more compelling approach, as one may see in the following sections, is Bayesian; it frames the problem of motor production as inference, and value function as beliefs. The *free energy* principle explains how actions follow from approximate Bayesian inference minimizing sensory surprise.

4.3 Free Energy and Active Inference

The free-energy principle [23] developed and popularised by Karl Friston attempts to be the unifying theory of the brain, perception and action. In this paradigm an agent is evolved to occupy some rather small subset of all the possible environment states. Note that agent's state is just a part of the full state of the environment. The environment evolves according to its stochastic dynamics, moving the agent away from their preferable set of states (natural habitat). The agent is equipped with a sensory system receiving $\tilde{s}$ and is also able to act on the environment. The goal of the agent is to minimize its surprise about the environment by in the following two ways: (a) adapting the sensory system through explaining the environment, which results in Bayesian inference on the causes of the sensory inputs, thus learning the model of the environment and making it less surprising; (b) acting on the environment to make it less surprising, thus resisting the natural increase of the entropy and moving to the preferred states.

Reformulating the above, the agent's internal state is a weighted compromise between the prior expectations about their ideal environment and information about the actual environment state provided through the sensory system; the weighting is done w.r.t. the uncertainties. The actions are performed to adjust the environment to fit the internal state. As the exact minimisation is intractable, a variational lower bound on surprise is used, which equals to free energy minimisation. Formally, in the most basic form, this results in the following optimization:

$$\mu_t = \arg\min_{\mu} F(\tilde{s}_t, \mu) \tag{4.11}$$

$$a_t = \pi(\tilde{s}, \mu) = \arg\min_{a} \mathbb{E}_{\tilde{s}_{t+1}|\tilde{s}_t,a} \left[F(\tilde{s}_{t+1}, \mu_{t+1})\right] \tag{4.12}$$

where $F(\tilde{s}, \mu)$ is the free energy lower bound on surprise having sensory state $\tilde{s}$, μ is the explanation of the sensory inputs (recognition density parameters), a is the emitted action. This approximation also gives rise to the compromise between the sensory adaptation and acting on the environment we mentioned before.

For continuous time models, optimization of imaginary sensory evolution vec-

tor field indirectly results in the optimal policy ([22]). The agent then produces actions to follow the path leading to minimization of free energy and eventually reaching the preferred distribution of sensory input.

Let's consider the following example: an anthropomorphic agent is dreaming of grasping a cup with a hand, as it may be a preferred (expected) state at some point. The cup is on the table, but the agent's internal state represents it almost grasped. A series of actions a_t are executed according to the equation (4.12), gradually minimizing the free energy until the cup is grasped. The sensory input expectations now match the actual sensory signal.

Active inference goes further on exploiting the Bayesian framework and recruits prior beliefs about motions and trajectories [60]. Such priors allows forming a much richer space of motor output than the cost function used in optimal control; according to Helmholtz decomposition, a sufficiently smooth vector field (policy representation) can be decomposed in a curl-free and a divergence-free components [21]. The curl-free part can be specified by the value function, while the divergence-free (periodic) requires more elaborate tricks like having path value, trajectory optimization, stationary state distribution prior [79].

While the Free Energy brain theory is very appealing and was able to give explanation to numerous processes in the brain, it lacks scalable implementations, action production was demonstrated only on toy examples, the desired state is explained as an optimization of free energy through the evolutionary process. As a very general principle it does not suggest or prefer any particular implementation, and can absorb any Bayesian approach.

4.4 Bayesian Perspective

Generating actions can be viewed as an inference in a Bayesian network representing an MPD. It is completely in agreement with the Free Energy principle, and can serve as its algorithmic level.

Continuous space discrete time Bayesian inference for MDP was suggested in [16] and rigorously formulated in [85] and extended in [34]. Toussaint [85] in-

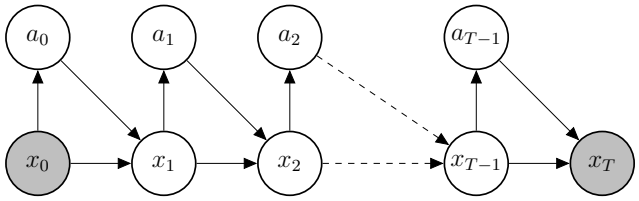


Figure 4.2: Graphical model for Markov decision process. State transitions $x_1 \dots x_{T-1}$ and actions $a_0 \dots a_{T-1}$ must be inferred having the Markov chain conditioned on the initial state x_0 and the goal x_T.

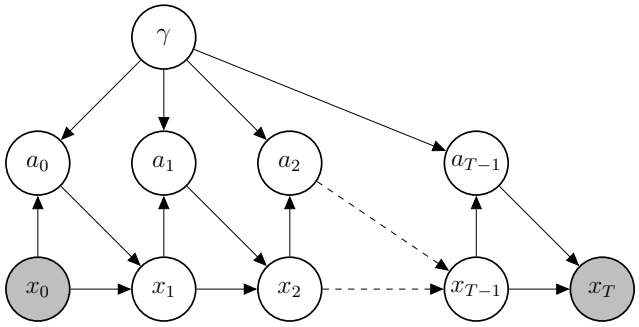


Figure 4.3: Bayesian Markov decision process. Function γ defines a prior over the controls $a_0 \dots a_{T-1}$. When sampled trajectories $\{x_0 \dots x_T\}$ are observed, inferring posterior of γ equals to inverse optimal control.

dicated that MAP estimate in a goal conditioned Markov chain results in solving the corresponding MDP (Fig. 4.2). Message passing algorithm [62], that provides efficient inference in Markov chains, combined with iterative expectation-maximization procedure for policy optimization results in a fast and robust solution for the MDP. This optimization procedure must be performed after every current observation x_c to account for the changes in the environment optimally:

$$a_c = \arg\max_{a_c} \prod_{t=c}^{T} p(x_{t+1}|x_t, a_t) \tag{4.13}$$

By setting a prior distribution on the control function, we arrive at the Bayesian optimal control (Fig. 4.3). Essentially, functional prior $p(a_0 \dots a_T|\gamma)$ defines a Markov random field, that in general case can be factorized as $\prod_{t=1}^{T-1} p(a_t|a_0 \dots a_{t-1}, \gamma)$. In this case, conditioning the full model on observed trajectories and performing Bayesian inference allows identifying the posterior of the control, giving a Bayesian solution to the Inverse Optimal Control. The control

generating function prior saves us from the problem of specifying the cost function; it also allows generating periodic (solenoidal) movements, which are required for gait, as well as non-periodic, like reaching and grasping.

The widely accepted framework of *forward and inverse models* in the brain [90], when viewed from the Bayesian perspective, boils down to explicit representation of the forward dynamics probabilities $p(x_{t+1}|x_t, a_t)$ and action posterior $p(a_t|x_{t+1}, x_t)$ in the brain, probably with multiple time scales. These probabilities can be learned form the experience and constantly being updated. The exact mechanism of learning and adapting is a subject of numerous experiments. It has been suggested that such forward model constitutes an intuitive physics model of the environment [41].

4.5 Motor Primitives

Just like in the visual system, where hierarchies of similar computations promote generalization, one may expect modularity in motor control [63]. Such modularity may be present on every layer of representation: muscle synergies, kinematics, joint dynamics, task grammar [20]. It is still not clear what exactly such "building blocks" are, but modularity in motor production and control can greatly simplify it (Fig. 4.4).

Motor output driving the muscles is not random; it has some structure. Attempts to find this structure, while following the biological constraints, produced a set of hypothesis about the generative process for the EMG, kinematic, and other motor data. Many motor primitive models are inspired by physiological studies of spinal cord and are mathematical models of CPGs and force fields.

A family of low-level motor primitives, called *synergies*, comes from the assumption that the motor output X is a linear combination of a small number of components (sources):

$$X = WS + \Sigma \tag{4.14}$$

$$\Sigma_{i,j} \sim \mathcal{N}(0, \sigma^2) \tag{4.15}$$

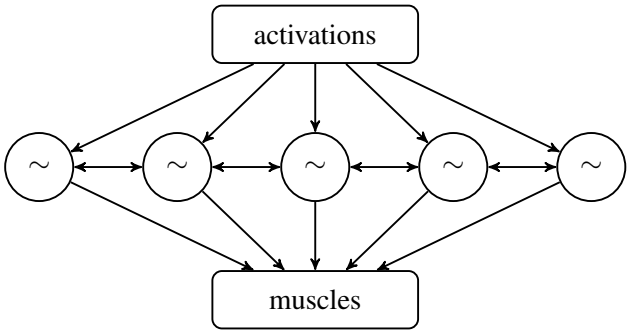


Figure 4.4: Motor primitives hypothesis. A set of learned pattern generators can be activated to produce a desirable motion. Equipped with feedback mappings they form feedback controllers. A small number of such motor primitives is sufficient to control a large task manifold.

where W - weights, S - concatenated sources, Σ - Gaussian noise. One can classify kinematic MPs into temporal, spatial, and spatio-temporal, depending on which dimensions the sources represent [18]. For temporal MPs, $S \in \mathbb{R}^{M,T}$ represents M temporal sources. Multiplied with $W \in^{D,M}$ we get D components for the full data $X \in \mathbb{R}^{D,T}$. In spatial MPs source matrix $S \in \mathbb{R}^{M,D}$ represents M sources of dimension D. The weights $W \in^{T,M}$ are time dependent. Correspondingly, in spatio-temporal MPs sources contain rich spatial and temporal activation patterns, essentially factorizing the data X similar to the Singular Value Decomposition (SVD).

Clearly, spatial synergies resemble the spinal force fields, while temporal activation components represent the spinal travelling waves, CPGs, or activations of the corresponding muscle synergies by motor cortex.

Analysis of kinematic data with component analysis (PCA), independent component analysis (ICA), anechoic mixture model (AMM), and smooth instantaneous mixtures (SIM) suggested AMM as the best candidate for such linear mixture. The representation of human gait trajectories with AMM requires only three components that can be shifted in time [18].

In the Temporal Motor Primitives model [11] motor output is created by activating a small set of sources that change their output in time. Linearly combining the sources with weights matrix produces the output. By changing the weights different outputs can be created. Learning the embedding of the weights allowed interpolation between the parameters of movements. When trained on data, generated

by an optimal control solver for a walking task, the TMPs could drive a humanoid robot to produce a parametric walk with different step sizes. It must be noted that the robot was walking without any feedback, thus TMPs running as a feed-forward controller.

Another type of movement primitives - Dynamic Motor Primitives (DMPs) - relies on learnable differential equations [35]. Controllers, constructed on top of differential equations with contracting property, guarantee the stability of generated motion. Combined with reinforcement learning methods, like Covariance Matrix Adaptation [70] or Path Integral Policy Iteration [39, 80], such MPs allowed for faster learning and modularity of the control. Thus we can conclude that real world reinforcement learning tasks benefit from parametric policies with good properties, and learned (sensori-)motor primitives may be interpreted as memories, storing optimal solutions. They promote learning and result in better generalization and avoid having to re-optimize we want to repeat an action [71, 11].

The type of motor primitives I suggested in my work is modular, as opposed to DMPs. Spatial synergies arise from the low dimensionality of the latent dynamics.

Chapter 5

Gaussian Process

The basic building block for our models of movement primitives is *Gaussian Process* (GP). In this chapter we review GP and its connection to linear regression and neural networks. GP is a non-parametric Bayesian method, which makes it particularly interesting for system identification applications: here non-parametric means that the number of parameters *grow* with the amount of training data, thus partially relieving the researcher from the problem of model search. Being a Bayesian method is also allows us to treat the uncertainty right. The drawback of GP is, however, its high training cost, as we explain later.

5.1 Linear Regression. Bayesian Perspective

Let's consider the following supervised machine learning problem: given training data points $X = \{x_i\}, Y = \{y_i\}, i \in 1 \dots N, x_i \in \mathbb{R}^D$ and fixed basis functions $B(x) = \{b_j(x)\}, j \in 1 \dots M$, find the regressor coefficients w_j assuming that observation noise is Gaussian:

$$\xi_i \sim \mathcal{N}(0, \sigma^2) \tag{5.1}$$

$$y_i = \sum_{j=1}^{M} w_j b_j(x_i) + \xi_i \tag{5.2}$$

Here we are interested in full Bayesian treatment of this problem (see e.g. [67]). By putting prior distributions on weights w and noise σ, we arrive to the graphical model at Fig. 5.1. By integrating out the parameters of the model, poste-

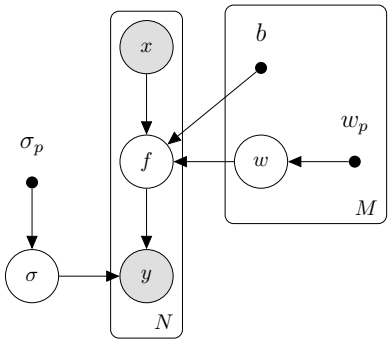


Figure 5.1: Bayesian linear regression. N noisy data points (x_i, y_i) are observed. The mapping $f(x)$ is approximated by weighted sum of M basis functions $b(x)$. To indicate multiple instantiations of parameters and random variables we used the standard plate notation [8].

rior predictive distribution for new inputs reads:

$$p(y^*|x^*, X, Y, \sigma) = \int p(y^*|x^*, X, Y, W, \sigma)p(W)dW \tag{5.3}$$

For simplicity, we put Gaussian prior with mean zero on W. Later, in the artificial neural network, this prior can be interpreted as a regulariser causing weights decay. With such prior the linear model gains in tractability. We may notice that prior predictive distribution is a huge Gaussian: as the prior on every weight is a Gaussian, every basis function value becomes a coefficient for these Gaussians, and then we just sum them up. As a sum of weighted normally distributed random variables is still a Gaussian, the combined prior on Y is a Gaussian.

$$p(w_i) = \mathcal{N}(0, 1) \tag{5.4}$$

$$p(w_i b_i(x_j)) = \mathcal{N}(0, b_i(x_j)^2) \tag{5.5}$$

$$p(y_j) = \mathcal{N}(0, \sum_{i=1}^{M} b_i(x_j)^2 + \sigma^2) \tag{5.6}$$

We will use the following equation to marginalize out normally distributed vari-

ables:

$$p(x) = \mathcal{N}(x|\mu, \Sigma) \tag{5.7}$$

$$p(y|x) = \mathcal{N}(y|Ax, \Lambda) \tag{5.8}$$

$$p(y) = \int p(y|x)p(x)dx \tag{5.9}$$

$$= \mathcal{N}(y|A\mu, A\Sigma A^T + \Lambda) \tag{5.10}$$

This integral in 5.9 can be carried out by a change of variables $z = Ax$ and interpreting it as a convolution. Two-sided Laplace transform [86] $\mathcal{B}\{\cdot\}(\cdot)$ allows us to sum up the cummulants of the Gaussian densities in the s-domain:

$$p(y) = \int \mathcal{N}(y|z, \Lambda)\mathcal{N}(A^{-1}z|\mu, \Sigma)\frac{1}{|\det(A)|}dz \tag{5.11}$$

$$= \int \mathcal{N}(y - z|0, \Lambda)\mathcal{N}(z|A\mu, A\Sigma A^T)dz \tag{5.12}$$

$$= \mathcal{B}^{-1}\left\{\mathcal{B}\{\mathcal{N}(y - z|0, \Lambda)\}(s)\mathcal{B}\{\mathcal{N}(z|A\mu, A\Sigma A^T)\}(s)\right\}(y) \tag{5.13}$$

$$= \mathcal{B}^{-1}\left\{\exp\left(\frac{1}{2}s^T\Lambda s\right)\exp\left(s^TA\mu + \frac{1}{2}s^TA\Sigma A^Ts\right)\right\}(y) \tag{5.14}$$

$$= \mathcal{B}^{-1}\left\{\exp\left(s^TA\mu + \frac{1}{2}s^T(A\Sigma A^T + \Lambda)s\right)\right\}(y) \tag{5.15}$$

$$= \mathcal{N}(y|A\mu, A\Sigma A^T + \Lambda) \tag{5.16}$$

Now we are equipped with right tools to integrate out W in 5.23:

$$W \in \mathbb{R}^M; W_i = w_i \tag{5.17}$$

$$X \in \mathbb{R}^{D\times N}; X_{:,i} = x_i \tag{5.18}$$

$$Y \in \mathbb{R}^N; Y_i = y_i \tag{5.19}$$

$$B \in \mathbb{R}^{M\times N}; B_{i,j} = b_i(x_j) \tag{5.20}$$

$$p(W) = \mathcal{N}(W|0, \mathbf{I}) \tag{5.21}$$

$$p(Y|X, W) = \mathcal{N}(B^T W, \mathbf{I}\sigma^2) \tag{5.22}$$

$$p(Y|X, \sigma) = \int p(Y, |X, W, \sigma)p(W)dW \tag{5.23}$$

$$= \mathcal{N}(0, B^T B + \mathbf{I}\sigma^2)) \tag{5.24}$$

Integrating out the weights resulted in a Gaussian distribution for Y. Notice that the covariance matrix $\mathbf{K}_{x,x} = B^T B + \mathbf{I}\sigma^2$, which we obtained by applying Eq. 5.10, is a function of inputs X. This matrix has a special name, Gram matrix, and defines the pairwise correlation structure of the outputs Y. Thus, we have arrived to *Gaussian process* - a stochastic process which outputs Y are normally distributed w.r.t inputs X (time, sample index, etc.). If outputs are normally distributed w.r.t. some covariance matrix, the posterior predictive is just a conditional Gaussian [67]

$$p(y^*|x^*, X, Y) = \mathcal{N}(y^*|\mathbf{K}_{x^*,x}\mathbf{K}_{x,x}^{-1}Y, \mathbf{K}_{x^*,x^*} - \mathbf{K}_{x^*,x}\mathbf{K}_{x,x}^{-1}\mathbf{K}_{x,x^*}) \tag{5.25}$$

From this equation we see the complexity of GP prediction - because of the Gram matrix inversion it is $O(N^3)$, which makes it suitable for datasets of order of no more than thousands of data points.

Also notice that every element of $B^T B$ matrix is a dot-product of the corresponding basis function values: $B^T B_{i,j} = [b_v(x_i)]^T[b_v(x_j)], v \in \{1 \dots M\}$. For a large enough number of basis functions computing the covariance matrix becomes unfeasible. We may, however, construct $B^T B$ directly, without the explicit mapping onto the basis functions (also often referred to as *feature space*). For this we use some results from functional analysis. We seek for such functions $k(\cdot, \cdot)$ of two

inputs which are equivalent to a dot-product of these inputs represented in some feature space. Such functions must meet the following condition (Mercer's) [73]:

$$k(x, x') = k(x', x) \tag{5.26}$$

$$\forall (f|f : \chi \to \mathbb{R}) : \int_{\chi \times \chi} f(x)k(x, x')f(x')dxdx' \geqslant 0 \tag{5.27}$$

This defines a positive (semi-)definite kernel $k(\cdot, \cdot)$, which is similar to the definition of positive (semi-)definite matrix, but operates in a continuous domain.

5.2 From Neural Network to Gaussian Process

Gaussian Process allows many interpretations: an infinite-dimensional Normal distribution (continuous limit of Eq. 5.24), a prior over functions, a non-linear regression etc. Here we clarify its connection to neural networks. To our knowledge, this connection was first made explicitly in [55] in the context of Bayesian treatment of artificial neural networks. The conclusion is that GPs with some kernel functions are computationally equivalent to a multilayered perceptron. This connection is relevant to my work as I used GPs to model neural network computations. Let's construct such network by introducing neurons with Radial Basis Function (RBF) receptive field:

$$RBF_{\mu}(x) = \exp\left(-\frac{1}{2}\frac{(x-\mu)^2}{l^2}\right) \tag{5.28}$$

We can always construct such receptive fields by adding responces of neurons with sigmoid activation functions according to the *universal approximation theorem* [14, 10]. The simplest approximation of the RBF can be achieved with only two sigmoid functions, i.e. two neurons:

$$RBF_{\mu}(x) \sim \frac{1}{1+\exp(x-\mu-1)} + \frac{1}{1+\exp(\mu-x-1)} - 1 \tag{5.29}$$

In our simple feed-forward network (Fig. 5.2) input vector I is fed into the hidden layer H. Every pair of neurons H_i has an Gaussian-like (RBF) receptive

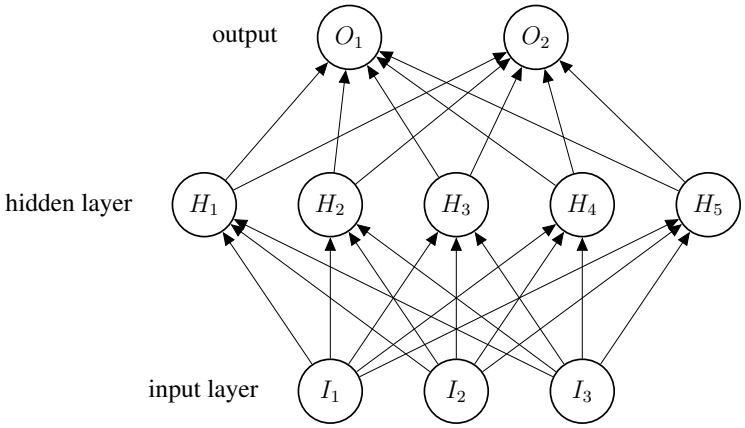


Figure 5.2: Feed-forward neural network with three layers.

field, with maxima equally spaced on some range of allowed inputs. The output layer O performs linear readout from the hidden units. We may notice that computationally this network is equivalent to the linear regression in some feature space: hidden units map inputs to a high-dimensional feature space, linear readout is just a linear combination of activated functions in that space. Again, as in the Bayesian regression example, we may compute the covariance values of the inputs x and x' defined by such network with M hidden neurons (Eq. 5.24) and RBF activation, which reads:

$$\phi(x) = \left[\exp\left(-\frac{1}{2}\frac{(x-\mu_i)^2}{l^2}\right)\right]_{i=1\ldots M} \tag{5.30}$$

$$\langle\phi(x), \phi(x')\rangle = \sum_{i=1}^{M} \exp\left(-\frac{1}{2}\frac{(x-\mu_i)^2}{l^2}\right)\exp\left(-\frac{1}{2}\frac{(x'-\mu_i)^2}{l^2}\right) \tag{5.31}$$

Complexity of computing activations of such network grows linearly with the number of hidden units, and for N data points the covariance matrix of activations requires $O(N * N * M)$ evaluations of RBF. By setting the number of hidden units to infinity, we may, however, perform a transition to the continuous domain and

compute such covariance matrix analytically:

$$\lim_{M\to\infty}\langle\phi(x),\phi(x')\rangle = \int_{-\infty}^{\infty}\exp\left(-\frac{1}{2}\frac{(x-\mu)^2}{l^2}\right)\exp\left(-\frac{1}{2}\frac{(x'-\mu)^2}{l^2}\right)d\mu \tag{5.32}$$

$$= \int_{-\infty}^{\infty}\exp\left(-\frac{1}{2}\frac{2(\mu-0.5(x+x'))^2}{l^2}\right)\exp\left(-\frac{1}{2}\frac{0.5(x-x')^2}{l^2}\right)d\mu \tag{5.33}$$

$$= Z\left(\frac{1}{2}l^2\right)\exp\left(-\frac{1}{2}\frac{(x-x')^2}{2l^2}\right) \tag{5.34}$$

where $Z\left(\frac{1}{2}l^2\right) = \sqrt{\pi l^2}$, normalizing constant for a Gaussian with covariance $\frac{1}{2}l^2$.

We see, that such RBF neural network in the limit of number of hidden units to infinity is equivalent to a GP with a similar looking kernel function. This integrated covariance function is known as ARD (Automatic Relevance Determination) kernel function. When trained, the influence of input dimensions on the output depends on the value of coefficients l_i. As l_i goes to infinity, the corresponding representation of the input in the feature space becomes smoother w.r.t. input dimension i and less relevant to the output.

A curious reader may ask what kinds of feature functions one can use to construct the covariance matrix. The answer is, essentially, any rich enough class of functions. Sometimes it is possible to make such transition to infinite number of functions as we did with RBFs and derive an analytic equation for the corresponding kernel (see, e.g. derivation of covariance function for sigmoidal neural networks in [87]). Also, one can use an arbitrarily complex (deep) neural network to construct a rich an flexible feature space, and then feed it into some kernel function to make it compatible with the GP framework ([88]).

In summary, we give a quick path from brain to GP: use artificial neural networks to model computations performed by biological neurons, add Bayesian treatment to handle uncertainty, find a class of networks where some of the Bayesian computations can be carried out analytically, which converge to GP in the limit of infinite capacity.

At this point the reader should be sufficiently equipped with the motivation

and links between the computational (GP) and implementational (neural network) levels of representation to justify the abstraction of the network architectures and working solely in more mathematical domain, like priors on functional mappings posed by GPs. This allows us to leverage the analytical machinery of mathematics with computational methods for optimization to achieve results in often intractable problems, while keeping the interpretation in terms of artificial neural networks always possible.

Bibliography

[4] Richard Bellman. *Dynamic programming*. en. OCLC: 830865530. Princeton, NJ: Princeton Univ. Pr, 1957.

[5] Nikolai A. Bernstein. *The Co-ordination and Regulation of Movements*. Oxford, New York, Pergamon Press, 1967.

[6] Dimitri P. Bertsekas. *Dynamic Programming and Optimal Control*. 1st. Athena Scientific, 1995.

[7] Patrick Billingsley. *Probability and measure*. en. 3rd ed. Wiley series in probability and mathematical statistics. New York: Wiley, 1995.

[8] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006.

[9] Christopher A. Buneo and Richard A. Andersen. "The posterior parietal cortex: Sensorimotor interface for the planning and online control of visually guided movements". en. In: *Neuropsychologia*. Visuomotor Functions of the Posterior Parietal Cortex 44.13 (Jan. 2006), pp. 2594–2606.

[10] Tianping Chen, Hong Chen, and Ruey-wen Liu. "A Constructive Proof and An Extension of Cybenko's Approximation Theorem". en. In: *Computing Science and Statistics*. Ed. by Connie Page and Raoul LePage. New York, NY: Springer New York, 1992, pp. 163–168.

[11] Debora Clever et al. "A novel approach for the generation of complex humanoid walking sequences based on a combination of optimal control and learning of movement primitives". en. In: *Robotics and Autonomous Systems* 83 (Sept. 2016), pp. 287–298.

[12] Anne Collins and Etienne Koechlin. “Reasoning, Learning, and Creativity: Frontal Lobe Function and Human Decision-Making”. en. In: *PLOS Biology* 10.3 (Mar. 2012), e1001293.

[13] Dylan F. Cooke and Michael S. A. Graziano. “Sensorimotor Integration in the Precentral Gyrus: Polysensory Neurons and Defensive Movements”. en. In: *Journal of Neurophysiology* 91.4 (Apr. 2004), pp. 1648–1660.

[14] G. Cybenko. “Approximation by superpositions of a sigmoidal function”. en. In: *Mathematics of Control, Signals and Systems* 2.4 (Dec. 1989), pp. 303–314.

[15] Andrea d’Avella et al. “Control of Fast-Reaching Movements by Muscle Synergy Combinations”. en. In: *Journal of Neuroscience* 26.30 (July 2006), pp. 7791–7810.

[16] Peter Dayan and Geoffrey E. Hinton. “Using Expectation-Maximization for Reinforcement Learning”. In: *Neural Computation* 9.2 (1997), pp. 271–278.

[17] Timothy J. Ebner and Siavash Pasalar. “Cerebellum Predicts the Future Motor State”. en. In: *The Cerebellum* 7.4 (Dec. 2008), pp. 583–588.

[18] Dominik M. Endres, Enrico Chiovetto, and Martin Giese. “Model selection for the extraction of movement primitives”. English. In: *Frontiers in Computational Neuroscience* 7 (2013).

[19] David Ferrier. “On the Localisation of the Functions of the Brain”. In: *British Medical Journal* 2.729 (Dec. 1874), pp. 766–767.

[20] Tamar Flash and Binyamin Hochner. “Motor primitives in vertebrates and invertebrates”. en. In: *Current Opinion in Neurobiology*. Motor sytems / Neurobiology of behaviour 15.6 (Dec. 2005), pp. 660–666.

[21] Karl J. Friston and Ping Ao. “Free Energy, Value, and Attractors”. In: *Comp. Math. Methods in Medicine* 2012 (2012), 937860:1–937860:27.

[22] Karl J. Friston, Jean Daunizeau, and Stefan J. Kiebel. “Reinforcement Learning or Active Inference?” en. In: *PLoS ONE* 4.7 (July 2009). Ed. by Olaf Sporns, e6421.

[23] Karl Friston, James Kilner, and Lee Harrison. “A free energy principle for the brain”. en. In: *Journal of Physiology-Paris* 100.1-3 (July 2006), pp. 70–87.

[24] A. P. Georgopoulos, A. B. Schwartz, and R. E. Kettner. “Neuronal population coding of movement direction”. en. In: *Science* 233.4771 (Sept. 1986), pp. 1416–1419.

[25] Michael Graziano. “The Organization of Behavioral Repertoire in Motor Cortex”. en. In: *Annual Review of Neuroscience* 29.1 (July 2006), pp. 105–134.

[26] Michael S. A. Graziano and Tyson N. Aflalo. “Mapping Behavioral Repertoire onto the Cortex”. en. In: *Neuron* 56.2 (Oct. 2007), pp. 239–251.

[27] Michael S. A. Graziano, Tyson N. S. Aflalo, and Dylan F. Cooke. “Arm Movements Evoked by Electrical Stimulation in the Motor Cortex of Monkeys”. In: *Journal of Neurophysiology* 94.6 (Dec. 2005), pp. 4209–4223.

[28] Michael S. A Graziano, Charlotte S. R Taylor, and Tirin Moore. “Complex Movements Evoked by Microstimulation of Precentral Cortex”. en. In: *Neuron* 34.5 (May 2002), pp. 841–851.

[29] S. Grillner. “Neurobiological bases of rhythmic motor acts in vertebrates”. en. In: *Science* 228.4696 (Apr. 1985), pp. 143–149.

[30] Sten Grillner and Peter Wallén. “Central Pattern Generators for Locomotion, with Special Reference to Vertebrates”. en. In: (1985), p. 29.

[31] Sten Grillner and Peter Zangger. “The effect of dorsal root transection on the efferent motor pattern in the cat’s hindlimb during locomotion”. en. In: *Acta Physiologica Scandinavica* 120.3 (1984), pp. 393–405.

[32] Joseph Y. Halpern. *Reasoning About Uncertainty*. Cambridge, MA, USA: MIT Press, 2003.

[33] Geoffrey E. Hinton. "Products of experts". In: *Artificial Neural Networks, 1999. ICANN 99. Ninth International Conference on (Conf. Publ. No. 470)*. Vol. 1. IET. 1999, pp. 1–6.

[34] Matthew Hoffman et al. "An Expectation Maximization Algorithm for Continuous Markov Decision Processes with Arbitrary Reward". In: *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*. Ed. by David van Dyk and Max Welling. Vol. 5. Proceedings of Machine Learning Research. Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA: PMLR, 16–18 Apr 2009, pp. 232–239.

[35] Auke Jan Ijspeert et al. "Dynamical Movement Primitives: Learning Attractor Models for Motor Behaviors". en. In: *Neural Computation* 25.2 (Feb. 2013), pp. 328–373.

[36] Kelvin E. Jones, Antonia F. de C. Hamilton, and Daniel M. Wolpert. "Sources of Signal-Dependent Noise During Isometric Force Production". en. In: *Journal of Neurophysiology* 88.3 (Sept. 2002), pp. 1533–1544.

[37] M. Kac. "On Distributions of Certain Wiener Functionals". In: *Transactions of the American Mathematical Society* 65.1 (1949), pp. 1–13.

[38] John F. Kalaska. "From Intention to Action: Motor Cortex and the Control of Reaching Movements". en. In: *Progress in Motor Control*. Ed. by Dagmar Sternad. Vol. 629. Boston, MA: Springer US, 2009, pp. 139–178.

[39] Hilbert J. Kappen. "Linear Theory for Control of Nonlinear Stochastic Systems". en. In: *Physical Review Letters* 95.20 (Nov. 2005), p. 200201.

[40] Mitsuo Kawato. "Internal models for motor control and trajectory planning". en. In: *Current Opinion in Neurobiology* 9.6 (Dec. 1999), pp. 718–727.

[41] Charles Kemp and Joshua B. Tenenbaum. "Structured statistical models of inductive reasoning." en. In: *Psychological Review* 116.1 (2009), pp. 20–58.

[42] David C. Knill and Whitman Richards, eds. *Perception As Bayesian Inference*. New York, NY, USA: Cambridge University Press, 1996.

[43] A.N. Kolmogorov. *Foundations of the theory of probability*. AMS Chelsea Publication. Chelsea Pub. Co., 1956.

[44] John W Krakauer and Pietro Mazzoni. "Human sensorimotor learning: adaptation, skill, and beyond". en. In: *Current Opinion in Neurobiology* 21.4 (Aug. 2011), pp. 636–644.

[45] Pierre Simon Laplace. *A Philosophical Essay on Probabilities*. Trans. by Frederick Truscott and Frederick Emory. Wiley, 1902.

[46] Rodolfo R. Llinas. *I of the Vortex: From Neurons to Self*. 1st ed. Bradford Books. The MIT Press, 2001.

[47] David J. C. MacKay. "Hyperparameters: Optimize, or Integrate Out?" en. In: *Maximum Entropy and Bayesian Methods*. Ed. by Glenn R. Heidbreder. Dordrecht: Springer Netherlands, 1996, pp. 43–59.

[48] G O Mackie and P Burighel. "The nervous system in adult tunicates: current research directions". en. In: *Canadian Journal of Zoology* 83.1 (Jan. 2005), pp. 151–183.

[49] D. Marr and T. Poggio. *From Understanding Computation to Understanding Neural Circuitry*. Tech. rep. Cambridge, MA, USA, 1976.

[50] C. D. Marsden, J. C. Rothwell, and B. L Day. "The use of peripheral feedback in the control of movement". en. In: *Trends in Neurosciences* 7.7 (July 1984), pp. 253–257.

[51] John H. Martin, Daniel Engber, and Zhuo Meng. "Effect of Forelimb Use on Postnatal Development of the Forelimb Motor Representation in Primary Motor Cortex of the Cat". en. In: *Journal of Neurophysiology* 93.5 (May 2005), pp. 2822–2831.

[52] R. C. Miall and D. M. Wolpert. “Forward Models for Physiological Motor Control”. en. In: *Neural Networks*. Four Major Hypotheses in Neuroscience 9.8 (Nov. 1996), pp. 1265–1279.

[53] Volodymyr Mnih et al. “Playing Atari with Deep Reinforcement Learning”. en. In: *arXiv:1312.5602 [cs]* (Dec. 2013). arXiv: 1312.5602.

[54] Pr Montague, P Dayan, and Tj Sejnowski. “A framework for mesencephalic dopamine systems based on predictive Hebbian learning”. en. In: *The Journal of Neuroscience* 16.5 (Mar. 1996), pp. 1936–1947.

[55] Radford M. Neal. *Bayesian Learning for Neural Networks*. Berlin, Heidelberg: Springer-Verlag, 1996.

[56] B. Øksendal. *Stochastic Differential Equations: An Introduction with Applications*. Hochschultext / Universitext. Springer, 2003.

[57] Bruno A. Olshausen and David J. Field. “Emergence of simple-cell receptive field properties by learning a sparse code for natural images”. en. In: *Nature* 381.6583 (June 1996), pp. 607–609.

[58] Mohsen Omrani et al. “Perspectives on classical controversies about the motor cortex”. en. In: *Journal of Neurophysiology* 118.3 (Sept. 2017), pp. 1828–1848.

[59] Simon A. Overduin et al. “Microstimulation Activates a Handful of Muscle Synergies”. en. In: *Neuron* 76.6 (Dec. 2012), pp. 1071–1077.

[60] Thomas Parr and Karl J. Friston. “Generalised free energy and active inference”. en. In: *Biological Cybernetics* (Sept. 2019).

[61] J. Pearl. “Bayesian Networks: A Model of Self-Activated Memory for Evidential Reasoning”. In: *Proc. of Cognitive Science Society (CSS-7)*. 1985.

[62] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1988.

[63] Tomaso Poggio and Emilio Bizzi. “Generalization in vision and motor control”. en. In: *Nature* 431.7010 (Oct. 2004), pp. 768–774.

[64] Aravind Rajeswaran et al. “Towards Generalization and Simplicity in Continuous Control”. en. In: *arXiv:1703.02660 [cs]* (Mar. 2018). arXiv: 1703.02660.

[65] D. Ramanathan, J. M. Conner, and M. H. Tuszynski. “A form of motor cortical plasticity that correlates with recovery of function after brain injury”. en. In: *Proceedings of the National Academy of Sciences* 103.30 (July 2006), pp. 11370–11375.

[66] Frank P. Ramsey. “Truth and Probability”. In: *The Foundations of Mathematics and other Logical Essays*. Ed. by R. B. Braithwaite. McMaster University Archive for the History of Economic Thought, 1926. Chap. 7, pp. 156–198.

[67] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.

[68] Jean-Alban Rathelot and Peter L. Strick. “Subdivisions of primary motor cortex based on cortico-motoneuronal cells”. en. In: *Proceedings of the National Academy of Sciences* 106.3 (Jan. 2009), pp. 918–923.

[69] J. C. Rothwell et al. “Manual Motor Performance in a Deafferented Man”. In: *Brain* 105.3 (Sept. 1982), pp. 515–542. eprint: `http://oup.prod.sis.lan/brain/article-pdf/105/3/515/955613/105-3-515.pdf`.

[70] Elmar A. Rückert et al. “Learned graphical models for probabilistic planning provide a new class of movement primitives”. English. In: *Frontiers in Computational Neuroscience* 6 (2013).

[71] Elmar Rückert and Andrea d’Avella. “Learned parametrized dynamic movement primitives with shared synergies for controlling robotic and musculoskeletal systems”. en. In: *Frontiers in Computational Neuroscience* 7 (2013).

[72] Philippe Saltiel et al. “Critical Points and Traveling Wave in Locomotion: Experimental Evidence and Some Theoretical Considerations”. English. In: *Frontiers in Neural Circuits* 11 (2017).

[73] Bernhard Scholkopf and Alexander J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA, USA: MIT Press, 2001.

[74] Wolfram Schultz. “Dopamine reward prediction error coding”. In: *Dialogues in Clinical Neuroscience* 18.1 (Mar. 2016), pp. 23–32.

[75] Stephen H. Scott. “Inconvenient Truths about neural processing in primary motor cortex”. en. In: *The Journal of Physiology* 586.5 (2008), pp. 1217–1224.

[76] Krishna V. Shenoy, Maneesh Sahani, and Mark M. Churchland. “Cortical Control of Arm Movements: A Dynamical Systems Perspective”. en. In: *Annual Review of Neuroscience* 36.1 (July 2013), pp. 337–359.

[77] Jeffrey C. Smith and Jack L. Feldman. “In vitro brainstem-spinal cord preparations for study of motor systems for mammalian respiration and locomotion”. en. In: *Journal of Neuroscience Methods* 21.2 (Oct. 1987), pp. 321–333.

[78] Richard S. Sutton and Andrew G. Barto. *Reinforcement learning: an introduction*. en. Second edition. Adaptive computation and machine learning series. Cambridge, Massachusetts: The MIT Press, 2018.

[79] Yuval Tassa, Tom Erez, and Emanuel Todorov. “Optimal Limit-Cycle Control recast as Bayesian Inference”. en. In: *IFAC Proceedings Volumes* 44.1 (Jan. 2011), pp. 4707–4713.

[80] E. Theodorou, J. Buchli, and S. Schaal. “Reinforcement learning of motor skills in high dimensions: A path integral approach”. In: *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. clmc. May 2010, pp. 2397–2403.

[81] Amanda S Therrien and Amy J Bastian. “Cerebellar damage impairs internal predictions for sensory and motor function”. en. In: *Current Opinion in Neurobiology*. Motor circuits and action 33 (Aug. 2015), pp. 127–133.

[82] A. N. Tikhonov and V. Y. Arsenin. *Solutions of Ill-posed problems*. W.H. Winston and sons, 1977.

[83] Emanuel Todorov and Michael I. Jordan. “A Minimal Intervention Principle for Coordinated Movement”. In: *Advances in Neural Information Processing Systems 15 [Neural Information Processing Systems, NIPS 2002, December 9-14, 2002, Vancouver, British Columbia, Canada]*. 2002, pp. 27–34.

[84] Emanuel Todorov and Michael I. Jordan. “Optimal feedback control as a theory of motor coordination”. en. In: *Nature Neuroscience* 5.11 (Nov. 2002), pp. 1226–1235.

[85] Marc Toussaint and Amos Storkey. “Probabilistic inference for solving discrete and continuous state Markov Decision Processes”. en. In: *Proceedings of the 23rd international conference on Machine learning - ICML ’06*. Pittsburgh, Pennsylvania: ACM Press, 2006, pp. 945–952.

[86] D. V. Widder. “What is the Laplace Transform?” In: *The American Mathematical Monthly* 52.8 (1945), pp. 419–425.

[87] Christopher K. I. Williams. “Computing with Infinite Networks”. In: *Advances in Neural Information Processing Systems 9*. Ed. by M. C. Mozer, M. I. Jordan, and T. Petsche. MIT Press, 1997, pp. 295–301.

[88] Andrew Gordon Wilson et al. “Deep Kernel Learning”. en. In: *arXiv:1511.02222 [cs, stat]* (Nov. 2015). arXiv: 1511.02222.

[89] D.M. Wolpert and M. Kawato. “Multiple paired forward and inverse models for motor control”. en. In: *Neural Networks* 11.7-8 (Oct. 1998), pp. 1317–1329.

[90] Daniel M Wolpert, R.Chris Miall, and Mitsuo Kawato. “Internal models in the cerebellum”. en. In: *Trends in Cognitive Sciences* 2.9 (Sept. 1998), pp. 338–347.

The Variational Coupled Gaussian Process Dynamical Model

Dmytro Velychko, Benjamin Knopp, and Dominik Endres

University of Marburg, Department of Psychology,
Gutenbergstr. 18, 35032 Marburg, Germany
{dmytro.velychko,benjamin.knopp,dominik.endres}@uni-marburg.de

Abstract. We present a full variational treatment of the Coupled Gaussian Process Dynamical Model (CGPDM) with non-marginalized coupling mappings. The CGPDM generates high-dimensional trajectories from coupled low-dimensional latent dynamical models. The deterministic variational treatment obviates the need for sampling and facilitates the use of the CGPDM on larger data sets. The non-marginalized coupling mappings allow for a flexible exchange of the constituent dynamics models at run time. This exchange possibility is crucial for the construction of modular movement primitive models. We test the model against the marginalized CGPDM, dynamic movement primitives and temporal movement primitives, finding that the CGPDM generally outperforms the other models. Human observers can hardly distinguish CGPDM-generated movements from real human movements.

Keywords: Gaussian Process, Variational Methods, Movement Primitives, Modularity

This is a preprint of the article
Velychko, D., Knopp B. and Endres, D.: The Variational Coupled Gaussian Process Dynamical Model. To be published in the Proceedings of the 26th International Conference on Artificial Neural Networks, 1-8 (2017).
The final publication will be available at DOI: 10.1007/978-3-319-68600-4 or http://www.springerlink.com

1 Introduction and Related Work

Planning and execution of human full-body movements is a formidable control problem for the brain. Modular movement primitives (MP) have been suggested as a means to simplify this control problem while retaining a sufficient degree of control flexibility for a wide range of task, see [4] for a review. 'Modular' in this context usually refers to the existence of an operation which allows for the combination of (simple) primitives into (complex) movements.

Technical applications of modular MPs have also been devised. For example in computer graphics, especially combined with dynamics models [7] and robotics,

e.g. the dynamical MP (DMP) [9]. Each DMP is encoded by a canonical second order differential equation with guaranteeable stability properties and learnable parameters.

To lift the restriction of canonical dynamics, the Coupled Gaussian Process Dynamical Model (CGPDM) [17] learns both the dynamics mappings and their coupling for a given movement. The learning is accomplished in a Gaussian process framework. The Gaussian process (GP) is a machine learning staple for classification and regression tasks. It can be interpreted as an abstraction of a neural network with a large, possibly infinite, hidden layer. Its advantages include theoretical elegance, tractability and closed-form solutions for posterior densities. It affords high flexibility but has poor (cubic) runtime scaling in the data set size. We improve this scaling with deterministic, sparse variational approximations using small sets of inducing points (IPs) and associated values [16] for each MP, resulting in the 'variational CGPDM' (vCGPDM). This yields a linear run-time dependence on the number of data points.

The CGPDM builds on the Gaussian process dynamical model (GPDM) [18], where a latent dynamics model is mapped onto observations by functions drawn from a GP. The GPDM can model the variability of human movements [15]. Sparse variational approximations have been developed for GPDM-like architectures [6] and even deep extensions thereof [11]. However, with the exception of the CGPDM, all these approaches have a 'monolithic' latent space(s) and thus lack the modularity of MPs. While deriving a variational approximation is not trivial, we expect it to avoid overfitting and yield a good bound on the marginal likelihood [2].

Our target application here is human movement modeling, but the vCGPDM could be easily applied to other systems where modularized control is beneficial, e.g. humanoid robotics [5].

We introduce the vCGPDM in section 2. In section 3, we first benchmark the vCGPDM against other MP models. Second, we determine the degree of human-tolerable sparseness in a psychophysics experiment. In section 4 we propose future research.

2 The model

A CGPDM is basically a number of GPDMs (the 'parts') run in parallel, with coupling between the latent space dynamics. See [17] for a graphical model representation. The model operates in discrete time $t = 0, \ldots, T$. For every part $i = 1, \ldots, M$ there is a Q^i-dimensional latent space with second-order autoregressive dynamics and inputs from the latent spaces of the other parts. Let $\boldsymbol{x}_t^i \in \mathbb{R}^{Q^i}$ be the state of latent space i at time t. Then

$$\boldsymbol{x}_t^i = \boldsymbol{f}^i(\boldsymbol{x}_{t-2}^1, \boldsymbol{x}_{t-1}^1, \ldots, \boldsymbol{x}_{t-2}^M, \boldsymbol{x}_{t-1}^M). \tag{1}$$

We chose a second-order model, because our target application is human movement modeling, and the literature indicates (e.g. [15]) that this is a good choice for this task. However, we note that this can be easily changed in the model. The

latent states $\boldsymbol{x}_t^i$ give rise to D^i-dimensional observations $\boldsymbol{y}_t^i \in \mathbb{R}^{D^i}$ via functions $\boldsymbol{g}^i(.)$ plus isotropic Gaussian noise η_t^i

$$\boldsymbol{y}_t^i = \boldsymbol{g}^i(\boldsymbol{x}_t^i) + \eta_t^i \tag{2}$$

The functions $\boldsymbol{g}^i(.)$ are drawn from a GP prior with zero mean function and a suitable kernel. In a vCGPDM, the functions $\boldsymbol{f}^i(\ldots)$ are also drawn from a GP prior with zero mean function, and and a kernel that is derived with product-of-experts (PoE, [8]) coupling between the latent spaces of the different parts, as described by [17]: each part generates a Gaussian prediction about every part (i.e. including itself). Let $\boldsymbol{x}_t^{i,j} = f^{i,j}(\boldsymbol{x}_{t-2}^i, \boldsymbol{x}_{t-1}^i)$ be the mean of the prediction of part i about part j at time index t, and $\alpha^{i,j}$ its variance. Following the standard PoE construction of multiplying the densities of the individual predictions and re-normalizing, one finds

$$p(\boldsymbol{x}_t^j | \boldsymbol{x}_t^{:,j}, \alpha^{:,j}) = \frac{\exp\left[-\frac{1}{2\alpha^j}\left(\boldsymbol{x}_t^j - \alpha^j \sum_i \frac{\boldsymbol{x}_t^{i,j}}{\alpha^{i,j}}\right)^2\right]}{(2\pi\alpha^j)^{\frac{Q^j}{2}}} \propto \prod_i \mathcal{N}\left(\boldsymbol{x}_t^j | \boldsymbol{x}_t^{i,j}, \alpha^{i,j}\right) \tag{3}$$

where $\alpha^j = \left(\sum_i \alpha_{i,j}^{-1}\right)^{-1}$. It was shown in [17] that the individual predictions $\boldsymbol{x}_t^{i,j}$ can be marginalized out in closed form. We will keep the individual predictions, because this allows us to couple a previously learned dynamics model for a part (including its predictions about the other parts) to any other dynamics model for the other parts, thus obtaining a modular MP model.

The form of eqn. 3 indicates the function of the coupling variances: the smaller a given variance, the more important the prediction of the generating part. When the $\alpha^{i,j}$ are optimized during learning, the model is able to discover which couplings are important for predicting the data, and which ones are not, see [17]. Put differently, if an $\alpha^{i,j}$ is small compared to $\alpha^{i' \neq i,j}$, then part i is able to make a prediction about part j with (relatively) high certainty. Furthermore, as demonstrated in [17], the $\alpha^{i,j}$ can be modulated after learning to generate novel movements which were not in the training data.

The basic CGPDM exhibits the usual cubic run time scaling with the number of data points, which prohibits learning from large data sets. We therefore developed a sparse variational approximation, following the treatment in [16,11]. We augment the model with IPs $\boldsymbol{r}^i$ and associated values $\boldsymbol{v}^i$ such that $g^i(\boldsymbol{r}^i) = \boldsymbol{v}^i$ for the latent-to-observed mappings $g^i(X^i)$ (referred to as 'LVM IPs' in the following), and condition the probability density of the function values of $g^i(X^i)$ on these points/values, which we assume to be a sufficient statistic. We apply the same augmentation strategy to reduce the computational effort for learning the dynamics mappings, which are induced by $\boldsymbol{z}^{i,j}$ and $\boldsymbol{u}^{i,j}$ (referred to as 'dynamics IP').

Key assumption of the vCGPDM: to obtain a tractable variational posterior distribution q over the latent states $\boldsymbol{x}_t^i = (x_{t,1}^i, \ldots, x_{t,Q^i}^i)$, we choose a distribution that factorizes across time steps $0, \ldots, T$, parts $1, \ldots, M$ and

dimensions $1, \ldots, Q^i$ within parts, and assume that the individual distributions are Gaussian:

$$q(\boldsymbol{x}_0^1, \ldots, \boldsymbol{x}_T^M) = \prod_{t=0}^{T} \prod_{i=1}^{M} \prod_{q=1}^{Q^i} q(\boldsymbol{x}_{t,q}^i) \; ; \quad q(\boldsymbol{x}_{t,q}^i) = \mathcal{N}(\mu_{t,q}^i, \sigma_{t,q}^{2,i}). \tag{4}$$

This approximation assumption is clearly a gross simplification of the correct latent state posterior. However, it allows us to make analytical progress: a free-energy evidence lower bound, ELBO (see eqn. 8 of [16] and eqn. S20 in the online supplementary material[1]) can now be computed in closed form if we choose the right kernels for the GPs. We opt for an ARD (automatic relevance detection) squared exponential kernel [3] for every part-i-to-j prediction GP:

$$k^{i,j}(\boldsymbol{X}, \boldsymbol{X}') = \exp\left(-\frac{1}{2} \sum_{q}^{Q^i} \frac{(\boldsymbol{X}_q - \boldsymbol{X}'_q)^2}{\lambda_q^{i,j}} \right). \tag{5}$$

and a radial basis function kernel for the latent-to-observed mappings. The computations yielding the ELBO are lengthy (and error-prone) but straightforward. The details can be found in section 2 of the online supplementary material. Whether our simplistic approximation assumption (eqn. 4) is useful depends on the data, but at least for human movement it seems appropriate (see section 3).

3 Results

We implemented the model in `Python 2.7` using the machine-learning framework `Theano` [1] for automatic differentiation to enable gradient-based maximization of the ELBO with the `scipy.optimize.fmin_l_bfgs_b` routine [10]. Latent space trajectories were initialized with PCA.

While the sparse approximations in the vCGPDM greatly reduce the memory consumption of the model, they might also introduce errors. Also, our fully factorized latent posterior approximation (eqn. 4) might be too simple. We tried to quantify these errors in a cross-validatory model comparison, and in a human perception experiment.

3.1 Human movement data

Comparisons were carried out on human movement data. We recorded these data with a 10-camera PhaseSpace Impulse motion capture system, mapped them onto a skeleton with 19 joints and computed joint angles in angle-axis representation, yielding a total of 60 degrees of freedom. The actors were instructed to walk straight with a natural arm swing, and to walk while waving both arms. Five walking-only and four walking+waving sequences each were used to train the models.

[1] available at `http://uni-marburg.de/wk8Vf`

3.2 MAP is worse than variational approximation

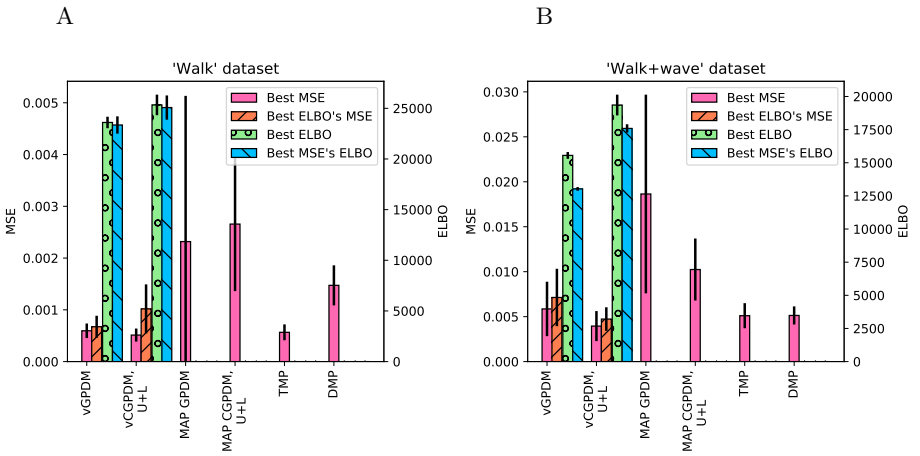


Fig. 1. Model comparison results. Shown is the average squared kinematics error on held-out data after dynamic time warping (MSE) and the variational lower bound on the model evidence (ELBO), where available. Error bars are standard errors of the mean. **A**: walking dataset. **B**: walking+waving dataset. For model descriptions and further details, see text.

To check how the predictive quality is affected by our sparse variational approximation, we conducted a comparison by five/four-fold cross-validation of the following models for walking/walking+waving. Our cross-validation score is the kinematics mean squared error (MSE), computed after dynamic time warping [14] of trajectories generated by initializing the model to the first two frames of a held-out trial onto the complete held-out trial: 1.) a GPDM with maximum-a-posteriori (MAP) estimation of the latent variables [18], called MAP GPDM in fig. 1. 2.) a fully marginalized two-part (upper/lower body) CGPDM with MAP estimation of the latent variables [17], called MAP CGPDM U+L. 3.) Their variational counterparts, vCGPDM U+L and vGPDM. We experimented with # LVM IPs$= 4, \ldots, 30$, and # dynamics IPs$= 2, \ldots, 30$. The MSE optima were near 10-15 IPs for both. All latent spaces were three-dimensional. 4.) Temporal movement primitives (instantaneous linear mixtures of functions of time) [5]. We used up to 10 primitives, the MSE optimum was located at ≈ 6. 5.) Dynamical movement primitives (DMP) [9]. We used between 1-50 basis functions, the lowest MSE was found at ≈ 15.

The results are plotted in fig. 1. Generally, all models perform better on the walking only dataset, than on walking+waving. This might be due to the latter being a more complex movement, as can be seen in the movie `modular_primitives.avi` in the online supplementary material. Of all tested models, the 2-part vCGPDM performs best in terms of MSE. It is significantly

better than the full-capacity (no IPs) MAP models, i.e. the development of a variational approximation which needs to store only ≈ 10 IPs rather than $\approx 10^4$ data points was well worth the effort. Furthermore, note that the Best ELBO's MSE (i.e. the MSE at the maximum of the ELBO w.r.t the #IPs) is a fairly good predictor of the best MSE, which indicates that our simple variational approximation is useful for model selection via ELBO. Further evidence for this is shown in fig. 1 of section 4 in the online supplementary material: we plotted MSE vs. ELBO for the vCGPDM U+L, symbols indicate different # LVM IPs. The negative correlation between MSE and ELBO is clearly visible. Furthermore, timing results for the vCGPDM can found in section 5 of the supplement, confirming the theoretical expectations of linear learning time scaling in the data set size for the vCGPDM.

Note that the vCGPDM U+L outperforms the vGPDM particularly on the 'walking+waving' dataset. This shows the usefulness of having modular, coupled dynamics models when the (inter)acting (body)parts execute partially independent movements. A visual demonstration of that modularity can be found in the video `modular_primitives.avi` in the online supplementary material.

3.3 A small number of IPs is enough to fool human observers

Next, we investigated the number of inducing points needed for perceptually plausible movements with a psychophysical experiment: We showed human observers ($n = 31$, 10 male, mean age: 23.8 ± 3.5a) videos of natural and artificial movements side-by-side on a computer screen. The artificial movements were generated by the vCGPDM U+L. After presentation, the participants had to choose the movement which they perceived as more natural. Examples of stimuli are provided in the online supplementary material in the movie `example_stimuli.mov` The walking sequences used for training and 9 additional walking sequences were used as natural stimuli. Each subject completed 1170 trials in randomized sequence, judging all artificial stimuli. We also tested for stimulus memorization effects via catch trials with previously unused natural movements in the last quarter of the experiment, finding none. All experimental procedures were approved by the local ethics commission.

Results are shown in fig. 2, A: we computed the frequency f_{gen} of choosing the vCGPDM-generated movement across all subjects as a function of the number of dynamics IPs and the number of LVM IPs. At best, we might expect f_{gen} to approach 0.5 when the generated movements are indistinguishable from the natural ones. We fitted those data with a logistic sigmoid $\frac{1}{1+\exp(a \cdot r(.)+c)}$ and a Bernoulli observation model, using two different regressor functions $r(.)$: a soft-minimum between the number of IPs and the MSE. Panel B shows the fit of f_{gen} with MSE, panel C shows 107-fold crossvalidation results for the two regressors, using the average negative log-probability on the held-out data as score. Error bars are standard deviations. 'Constant' is the constant regressor, any other regressor should predict better. 'Data' uses the data mean of the individual #IP combinations as a predictor, and constitutes a lower bound on the cross validation score.

Clearly, f_{gen} increases with the number of IPs, approaching (but not quite reaching) 0.5 for a sufficiently large number of IPs, this is true for the MSE regression, too. Hence, MSE is a good predictor of perceptual performance. Furthermore, a rather small number of IPs is sufficient for modeling this data. This allows for compactly parametrized MPs.

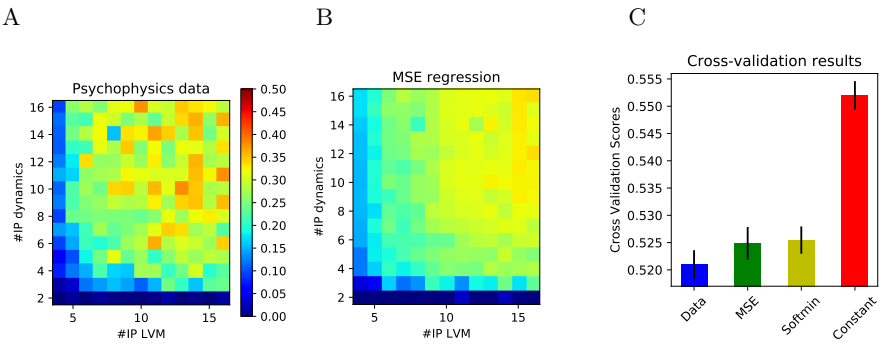


Fig. 2. Perceived naturalness of the model, as a function of the number of inducing points (#IP) **A**: Rate of perceiving vCGPDM-generated stimulus as more natural than natural stimulus, averaged across all participants. **B**: Regression of data in panel A, MSE as regressor and logistic sigmoid as psychometric function. **C**: Regression model comparison with 107-fold cross-validation. Softmin and MSE perform comparably well. Both are close to optimal.

4 Conclusion

We developed a full variational approximation of the CGPDM, the vCGPDM, which obviates the need for sampling the latent space trajectories [6]. We demonstrated that the vCGPDM with a small number of IPs performs better than the full-capacity CGPDM with a MAP approximation to the latent states, and that the vCGPDM is also able to outperform other contemporary MP models, most likely due to its learnable dynamics. Next, we showed that it produces perceptually believable full-body movements. While perceptual evaluations of full and sparse GPDM-like models [15] have been done before, we are the first to investigate systematically the number of IPs of all model components required for perceptual plausibility. Furthermore, we showed that the MSE and the number of IPs can be used to predict average human classification performance almost optimally. This indicates that the model selection process on large databases of training movements for the model could possibly be automated.

We are now in a position to learn a large library of movements with a CGPDM, and study its compositionality. This is possible due to the compact representation of each MP. Instead of direct connections between parts in the vCGPDM, it is also conceivable to embed the parts into a hierarchical architecture, like [15].

While the vCGPDM is suitable when the number of parts is relatively small (computational complexity $\mathcal{O}(T * M * (M * \#IP)^3)$ per optimization iteration), a hierarchical architecture might enable more computational savings for many parts. A further direction of future research are *sensorimotor* primitives, i.e. MPs that can be conditioned on sensory input [12,11,13] which we will implement by adding sensory predictions to the latent-to-observed mappings. **Acknowledgements** DFG-IRTG 1901 'The Brain in Action', DFG-SFB-TRR 135 project C06. We thank Olaf Haag for help with rendering the movies, and Björn Büdenbender for assistance with MoCap.

References

1. Bastien, F., Lamblin, P., Pascanu, R., Bergstra, J., Goodfellow, I.J., Bergeron, A., Bouchard, N., Bengio, Y.: Theano: new features and speed improvements. Deep Learning and Unsupervised Feature Learning NIPS Workshop (2012)
2. Bauer, M., van der Wilk, M., Rasmussen, C.: Understanding probabilistic sparse Gaussian process approximations. Tech. rep., arXiv:1606.04820 (2016)
3. Bishop, C.M.: Pattern Recognition and Machine Learning (Information Science and Statistics). Springer-Verlag New York, Inc., Secaucus, NJ, USA (2006)
4. Bizzi, E., Cheung, V., d'Avella, A., Saltiel, P., Tresch, M.: Combining modules for movement. Brain Res. Rev. 57(1), 125 – 133 (2008)
5. Clever, D., Harant, M., Koch, K.H., Mombaur, K., Endres, D.M.: A novel approach for the generation of complex humanoid walking sequences based on a combination of optimal control and learning of movement primitives. Rob. Aut. Sys. 83, 287–298 (2016), doi: 10.1016/j.robot.2016.06.001
6. Frigola, R., Chen, Y., Rasmussen, C.: Variational Gaussian process state-space models. In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., Weinberger, K. (eds.) Advances in NIPS 27, pp. 3680–3688 (2014)
7. Giese, M.A., Mukovskiy, A., Park, A.N., Omlor, L., Slotine, J.J.E.: Real-Time Synthesis of Body Movements Based on Learned Primitives. In Cremers D., Rosenhahn B., Yuille A. L. (eds): Statistical and Geometrical Approaches to Visual Motion Analysis, LNCS 5604, 107–127 (2009)
8. Hinton, G.E.: Products of experts. In: Proc. ICANN'99. vol. 1, pp. 1–6 (1999)
9. Ijspeert, A.J., Nakanishi, J., Hoffmann, H., Pastor, P., Schaal, S.: Dynamical movement primitives: Learning attractor models for motor behaviors. Neu. Comp. 25(2), 328–373 (2013)
10. Jones, E., Oliphant, T., Peterson, P., et al.: SciPy: Open source scientific tools for Python (2001–), `http://www.scipy.org/`, [Online; accessed 2015-10-09]
11. Mattos, C.L.C., Dai, Z., Damianou, A., Forth, J., Barreto, G.A., Lawrence, N.D.: Recurrent Gaussian processes. Tech. rep., arXiv:1511.06644 (2016)
12. Paraschos, A., Daniel, C., Peters, J., Neumann, G.: Probabilistic movement primitives. In: Burges, C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K. (eds.) Advances in NIPS 26, pp. 2616–2624 (2013)
13. Pastor, P., Kalakrishnan, M., Righetti, L., Schaal, S.: Towards associative skill memories. In: IEEE-RAS Conf. Humanoids. pp. 309–315 (2012)
14. Sakoe, H., Chiba, S.: Dynamic programming algorithm optimization for spoken word recognition. IEEE Trans. Acoust. Speech Sig. Proc. 26(1), 43–49 (Feb 1978)

15. Taubert, N., Christensen, A., Endres, D., Giese, M.: Online simulation of emotional interactive behaviors with hierarchical Gaussian process dynamical models. In: Proc. ACM SAP. pp. 25–32. ACM (2012)
16. Titsias, M.K., Lawrence, N.D.: Bayesian Gaussian process latent variable model. In: Proc. 13th AISTATS. pp. 844–851 (2010)
17. Velychko, D., Endres, D., Taubert, N., Giese, M.A.: Coupling Gaussian process dynamical models with product-of-experts kernels. In: Proc. 24th ICANN, LNCS 8681, pp. 603–610. Springer (2014)
18. Wang, J.M., Fleet, D.J., Hertzmann, A.: Gaussian process dynamical models for human motion. IEEE Trans. Pattern Anal. Mach. Intell. 30(2), 283–298 (2008)

Supplementary Material: The Variational Coupled Gaussian Process Dynamical Model

Dmytro Velychko, Benjamin Knopp, Dominik Endres*

December 10, 2019

1 Partial optimization of variational distribution for simplified ELBO

While optimizing the full variational posterior in augmented Gaussian Processes models the following type of term appears often in the ELBO equation:

$$\begin{aligned}\mathcal{R}(q(\boldsymbol{u}), q(\boldsymbol{v})) &= \int q(\boldsymbol{u})(f(q(\boldsymbol{v}), \boldsymbol{u}) + \log \frac{p(\boldsymbol{u})}{q(\boldsymbol{u})})d\boldsymbol{u} \\ &= \int q(\boldsymbol{u}) f(q(\boldsymbol{v}), \boldsymbol{u})d\boldsymbol{u} + \int q(\boldsymbol{u}) \log p(\boldsymbol{u})d\boldsymbol{u} - \int q(\boldsymbol{u}) \log q(\boldsymbol{u})d\boldsymbol{u}\end{aligned} \tag{S1}$$

To simplify the optimization of such terms, we would like to carry out the optimization with respect to the density $q(\boldsymbol{u})$ analytically, so as to remove the dependency on $q(\boldsymbol{u})$. To this end, we calculate for the optimal variational $q^*(\boldsymbol{u})$ in the above equation. This approach was suggested in [1], however, it is not well described there. Here we give an extended derivation. A necessary condition for maximality is a vanishing functional derivative under the constraint that the density $q(\boldsymbol{u})$ is normalized to one:

$$\int q(\boldsymbol{u})d\boldsymbol{u} - 1 = 0 \tag{S2}$$

which is fulfilled at the stationary points of the Lagrangian

$$\mathcal{X}(q(\boldsymbol{u}), q(\boldsymbol{v})) = \mathcal{R}(q(\boldsymbol{u}), q(\boldsymbol{v})) + \lambda \left(\int q(\boldsymbol{u})d\boldsymbol{u} - 1 \right) \tag{S3}$$

*Theoretical Neuroscience Group, Dept. Psychology, University of Marburg, Gutenbergstr. 18, 35032 Marburg, Germany
{dmytro.velychko,benjamin.knopp,dominik.endres}@uni-marburg.de

where λ is chosen so that (S2) holds. Taking the derivative of $\mathcal{X}(q(\boldsymbol{u}), q(\boldsymbol{v}))$ and setting it to zero yields

$$\frac{\delta \mathcal{X}(q(\boldsymbol{u}), q(\boldsymbol{v}))}{\delta q(\boldsymbol{u})} = f(q(\boldsymbol{v}), \boldsymbol{u}) + \log p(\boldsymbol{u}) - \log q(\boldsymbol{u}) + 1 + \lambda = 0 \tag{S4}$$

and therefore, denoting $Z = \exp(-\lambda - 1)$

$$q^*(\boldsymbol{u}) = \exp(f(q(\boldsymbol{v}), \boldsymbol{u}) + \log p(\boldsymbol{u}) + 1 + \lambda) \tag{S5}$$

$$q^*(\boldsymbol{u}) = \frac{1}{Z} p(\boldsymbol{u}) \exp(f(q(\boldsymbol{v}), \boldsymbol{u})) \tag{S6}$$

$$Z = \exp(-\lambda - 1) = \int p(\boldsymbol{u}) \exp(f(q(\boldsymbol{v}), \boldsymbol{u})) d\boldsymbol{u} \tag{S7}$$

Substituting the optimal $q^*(\boldsymbol{u})$ into the original term, we get:

$$\begin{aligned}
\mathcal{R}(q(\boldsymbol{v})) &= \int \frac{1}{Z} p(\boldsymbol{u}) \exp(f(q(\boldsymbol{v}), \boldsymbol{u}))(f(q(\boldsymbol{v}), \boldsymbol{u}) + \log \frac{p(\boldsymbol{u})}{\frac{1}{Z} p(\boldsymbol{u}) \exp(f(q(\boldsymbol{v}), \boldsymbol{u}))}) d\boldsymbol{u} \\
&= \int \frac{1}{Z} p(\boldsymbol{u}) \exp(f(q(\boldsymbol{v}), \boldsymbol{u})) \left(\log \frac{p(\boldsymbol{u}) \exp(f(q(\boldsymbol{v}), \boldsymbol{u}))}{\frac{1}{Z} p(\boldsymbol{u}) \exp(f(q(\boldsymbol{v}), \boldsymbol{u}))} \right) d\boldsymbol{u} \\
&= \log(Z) \frac{1}{Z} \int p(\boldsymbol{u}) \exp(f(q(\boldsymbol{v}), \boldsymbol{u})) d\boldsymbol{u} \\
&= \log \int p(\boldsymbol{u}) \exp(f(q(\boldsymbol{v}), \boldsymbol{u})) d\boldsymbol{u}
\end{aligned} \tag{S8}$$

This is the simplified version of (S1), which depends only on $q(\boldsymbol{v})$ because $q^(\boldsymbol{u}) = q^*(\boldsymbol{u})$ has been determined by optimization.

2 vCGPDM ELBO derivation

Here we give the extended derivation of the ELBO from the main paper. Let's assume we deal with M parts. The model reads:

$$\boldsymbol{X} = \{\boldsymbol{X}^0 \dots \boldsymbol{X}^M\} \tag{S9}$$

$$\boldsymbol{X}^i = \{\boldsymbol{x}_0^i \dots \boldsymbol{x}_T^i\}; \boldsymbol{x}_t^i \in \mathbb{R}^{Q^i} \tag{S10}$$

$$\boldsymbol{Y}^i = \{\boldsymbol{y}_0^i \dots \boldsymbol{y}_T^i\}; \boldsymbol{y}_t^i \in \mathbb{R}^{D^i} \tag{S11}$$

$$f^{j,i}(\boldsymbol{X}_{-t}^j) \sim \mathcal{GP}(0, k_f^j(\boldsymbol{x}_{-t}^j, \boldsymbol{x}_{-t}^{j\prime})) \tag{S12}$$

$$p(\boldsymbol{x}_t^i) \sim \prod_{j=1}^{M} \mathcal{N}(\boldsymbol{x}_t^i | f^{j,i}(\boldsymbol{x}_{-t}^j), \boldsymbol{I}\alpha^{j,i}); \alpha > 0 \tag{S13}$$

$$g_d^i(\boldsymbol{X}^i) \sim \mathcal{GP}(0, k_g(\boldsymbol{x}_t^i, \boldsymbol{x}_t^{i\prime})) \tag{S14}$$

$$\boldsymbol{g}_d^i = g_d^i(\boldsymbol{X}^i) \tag{S15}$$

$$\boldsymbol{Y}_{:,d}^i \sim \mathcal{N}(\boldsymbol{g}_d^i, \boldsymbol{I}\beta^i); \beta > 0 \tag{S16}$$

$$p(\boldsymbol{x}_0^i) = \mathcal{N}(\boldsymbol{x}_0^i | 0, \boldsymbol{I}) \tag{S17}$$

Here and following upper indexes are part-related, lower indexes indicate dimensions.

We have $M \times M$ latent dynamics mappings, which are combined into M mappings with the Product of Experts - multiplying and renormalizing the distribution from separate partial predictions. Each of the $M \times M$ mappings is augmented with some inducing inputs and outputs: $f^{j,i} : \boldsymbol{z}^{j,i} \to \boldsymbol{u}^{j,i}$. The full augmented joint distribution of the model reads:

$$
\begin{aligned}
p(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{f}, \boldsymbol{v}, \boldsymbol{g}, \boldsymbol{y}) &= p(\boldsymbol{y}|\boldsymbol{g}) p(\boldsymbol{g}|\boldsymbol{x}, \boldsymbol{v}) p(\boldsymbol{v}) p(\boldsymbol{x}, \boldsymbol{f}|\boldsymbol{u}) p(\boldsymbol{u}) \\
&= \left[\prod_{i=1}^{M} p(\boldsymbol{y}^i|\boldsymbol{g}^i) p(\boldsymbol{g}^i|\boldsymbol{x}^i, \boldsymbol{v}^i) p(\boldsymbol{v}^i) \right] p(\boldsymbol{x}, \boldsymbol{f}|\boldsymbol{u}) p(\boldsymbol{u}) \\
&= \left[\prod_{i=1}^{M} \left[\prod_{d=1}^{D_i} p(\boldsymbol{y}_d^i|\boldsymbol{g}_d^i) p(\boldsymbol{g}_d^i|\boldsymbol{x}^i, \boldsymbol{v}^i) \right] p(\boldsymbol{v}^i) \right] \\
&\times \left[\prod_{t=1}^{T} \left[\prod_{i=1}^{M} p(\boldsymbol{x}_t^i | \{\boldsymbol{f}_t^{:,i}, \alpha^{:,i}\}) \right] \left[\prod_{i=1}^{M} \prod_{j=1}^{M} p(\boldsymbol{f}_t^{j,i} | \boldsymbol{f}_{1:t-1}^{j,i}, \boldsymbol{x}_{0:t-1}^{j}, \boldsymbol{u}^{j,i}) \right] \right] \\
&\times \left[\prod_{i=1}^{M} \prod_{j=1}^{M} p(\boldsymbol{u}^{j,i}) \right] \left[\prod_{i=1}^{M} p(\boldsymbol{x}_0^i) \right]
\end{aligned} \tag{S18}
$$

The full proposal variational posterior is:

$$
\begin{aligned}
q(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{f}, \boldsymbol{v}, \boldsymbol{g}) &= p(\boldsymbol{g}|\boldsymbol{x}, \boldsymbol{v}) q(\boldsymbol{v}) p(\boldsymbol{f}|\boldsymbol{x}, \boldsymbol{u}) q(\boldsymbol{x}) q(\boldsymbol{u}) \\
&= p(\boldsymbol{g}|\boldsymbol{x}, \boldsymbol{v}) q(\boldsymbol{v}) \left[\prod_{t=1}^{T} \prod_{i=1}^{M} \prod_{j=1}^{M} p(\boldsymbol{f}_t^{j,i} | \boldsymbol{f}_{1:t-1}^{j,i}, \boldsymbol{x}_{0:t-1}^{j}, \boldsymbol{u}^{j,i}) \right] q(\boldsymbol{x}) q(\boldsymbol{u})
\end{aligned} \tag{S19}
$$

The ELBO is:

$$\mathcal{L}(\boldsymbol{\theta}) = \int_{\boldsymbol{x},\boldsymbol{v},\boldsymbol{g}} q(\boldsymbol{x},\boldsymbol{u},\boldsymbol{f},\boldsymbol{v},\boldsymbol{g}) \log\left(\frac{p(\boldsymbol{x},\boldsymbol{u},\boldsymbol{f},\boldsymbol{v},\boldsymbol{g},\boldsymbol{y})}{q(\boldsymbol{x},\boldsymbol{u},\boldsymbol{f},\boldsymbol{v},\boldsymbol{g})}\right) \tag{S20}$$

$$= \sum_{i=1}^{M}\sum_{d=1}^{D} \int_{\boldsymbol{x}^i,\boldsymbol{v}^i,\boldsymbol{g}_d^i} p(\boldsymbol{g}_d^i|\boldsymbol{x}^i,\boldsymbol{v}^i) q(\boldsymbol{x}^i) q(\boldsymbol{v}^i) \log\frac{p(\boldsymbol{y}_d^i|\boldsymbol{g}_d^i)}{q(\boldsymbol{v}^i)} \tag{S21}$$

$$+ \int q(\boldsymbol{u}) \left[\sum_{t=1}^{T} \int q(\boldsymbol{x}_t) q(\boldsymbol{x}_{-t}) \left(\int \left[\prod_{i=1}^{M}\prod_{j=1}^{M} p(\boldsymbol{f}_t^{j,i}|\boldsymbol{f}_{1:t-1}^{j,i}, \boldsymbol{x}_{0:t-1}^{j}, \boldsymbol{u}^{j,i})\right]\right.\right.$$

$$\left.\left.\times \log\prod_{i=1}^{M} p(\boldsymbol{x}_t^i|\{\boldsymbol{f}_t^{:,i}, \alpha^{:,i}\}) d\boldsymbol{f}_t\right) d\boldsymbol{x}_t d\boldsymbol{x}_{-t}\right]$$

$$+ q(\boldsymbol{u}) \log\frac{p(\boldsymbol{u})}{q(\boldsymbol{u})} d\boldsymbol{u}$$

$$+ \int q(\boldsymbol{x}_0) \log p(\boldsymbol{x}_0) d\boldsymbol{x}_0 + H(q(\boldsymbol{x})) \tag{S22}$$

The (S21) part is the GPLVM ELBO and is given in [2]. Next, we consider only the ELBO component which is relevant for the dynamics and apply the sufficient statistics assumption: knowing $\boldsymbol{x}_{-t}^j$ and $\boldsymbol{u}^{j,i}$ is sufficient for the $\boldsymbol{f}_t^{j,i}$ distribution. The innermost integral is:

$$\begin{aligned}
\mathcal{A} &= \int \left[\prod_{i=1}^{M}\prod_{j=1}^{M} p(\boldsymbol{f}_t^{j,i}|\boldsymbol{x}_{-t}^j, \boldsymbol{u}^{j,i})\right] \log\prod_{i=1}^{M} p(\boldsymbol{x}_t^i|\{\boldsymbol{f}_t^{:,i}, \alpha^{:,i}\}) d\boldsymbol{f}_t \\
&= \sum_{i=1}^{M} \int \left[\prod_{j=1}^{M} p(\boldsymbol{f}_t^{j,i}|\boldsymbol{x}_{-t}^j, \boldsymbol{u}^{j,i})\right] \log p(\boldsymbol{x}_t^i|\{\boldsymbol{f}_t^{:,i}, \alpha^{:,i}\}) d\boldsymbol{f}_t^i \\
&= \sum_{i=1}^{M} \int \left[\prod_{j=1}^{M} \mathcal{N}(\boldsymbol{f}_t^{j,i}|\boldsymbol{\mu}_{\boldsymbol{f}_t^{j,i}}, \boldsymbol{S}_{\boldsymbol{f}_t^{j,i}})\right] \log\mathcal{N}(\boldsymbol{x}_t^i|\alpha_i \sum_{j=1}^{M} \alpha_{j,i}^{-1} \boldsymbol{f}_t^{j,i}, \boldsymbol{I}\alpha_i) d\boldsymbol{f}_t^i \\
&= \sum_{i=1}^{M} \left[-\frac{1}{2}\operatorname{tr}\left[\alpha_i \sum_{j=1}^{M} (\alpha_{j,i}^{-1})^2 \boldsymbol{S}_{\boldsymbol{f}_t^{j,i}}\right] + \log\mathcal{N}(\boldsymbol{x}_t^i|\alpha_i \sum_{j=1}^{M} \alpha_{j,i}^{-1} \boldsymbol{\mu}_{\boldsymbol{f}_t^{j,i}}, \boldsymbol{I}\alpha_i)\right]
\end{aligned} \tag{S23}$$

$$\boldsymbol{\mu}_{\boldsymbol{f}_t^{j,i}} = \boldsymbol{K}_{\boldsymbol{x}_{-t}^j, \boldsymbol{z}^{j,i}}^{j,i} \left(\boldsymbol{K}_{\boldsymbol{z}^{j,i}, \boldsymbol{z}^{j,i}}^{j,i}\right)^{-1} \boldsymbol{u}^{j,i} \tag{S24}$$

$$\boldsymbol{S}_{\boldsymbol{f}_t^{j,i}} = \boldsymbol{K}_{\boldsymbol{x}_{-t}^j, \boldsymbol{x}_{-t}^j}^{j,i} - \boldsymbol{K}_{\boldsymbol{x}_{-t}^j, \boldsymbol{z}^{j,i}}^{j,i} \left(\boldsymbol{K}_{\boldsymbol{z}^{j,i}, \boldsymbol{z}^{j,i}}^{j,i}\right)^{-1} \boldsymbol{K}_{\boldsymbol{x}_{-t}^j, \boldsymbol{z}^{j,i}}^{j,i} \tag{S25}$$

$$\alpha_i = (\sum_{j=1}^{M} \alpha_{j,i}^{-1})^{-1} \tag{S26}$$

Now let's take the integral over $\boldsymbol{x}$:

$$\begin{aligned}
\mathcal{B} &= \int q(\boldsymbol{x}_t)q(\boldsymbol{x}_{-t})\mathcal{A}d\boldsymbol{x}_t d\boldsymbol{x}_{-t} \\
&= \sum_{i=1}^{M}\Bigg(-\frac{1}{2}\alpha_i\sum_{j=1}^{M}(\alpha_{j,i}^{-1})^2\,\mathrm{tr}\left[\Psi_0^{j,i}(\boldsymbol{x}_{-t}^j) - \left(\boldsymbol{K}_{\boldsymbol{z}^{j,i},\boldsymbol{z}^{j,i}}^{j,i}\right)^{-1}\Psi_2^{j,i}(\boldsymbol{x}_{-t}^j)\right] \\
&- \log Z(\boldsymbol{I}\alpha_i) - \frac{1}{2}\alpha_i^{-1}\left[\mathrm{tr}(\boldsymbol{S}_{\boldsymbol{x}_t^i}) + \boldsymbol{\mu}_{\boldsymbol{x}_t^i}^T\boldsymbol{\mu}_{\boldsymbol{x}_t^i}\right] \\
&+ \boldsymbol{\mu}_{\boldsymbol{x}_t^i}^T\Big(\sum_{j=1}^{M}\alpha_{j,i}^{-1}\Psi_1^{j,i}(\boldsymbol{x}_{-t}^j)\left(\boldsymbol{K}_{\boldsymbol{z}^{j,i},\boldsymbol{z}^{j,i}}^{j,i}\right)^{-1}\boldsymbol{u}^{j,i}\Big) \\
&- \frac{1}{2}\alpha_i\sum_{j=1}^{M}\sum_{k=1}^{M}\alpha_{j,i}^{-1}\alpha_{k,i}^{-1}\boldsymbol{u}^{j,iT}\left(\boldsymbol{K}_{\boldsymbol{z}^{j,i},\boldsymbol{z}^{j,i}}^{j,i}\right)^{-1}\Psi_2^{j,k,i}(\boldsymbol{x}_{-t}^j,\boldsymbol{x}_{-t}^k)\left(\boldsymbol{K}_{\boldsymbol{z}^{k,i},\boldsymbol{z}^{k,i}}^{k,i}\right)^{-1}\boldsymbol{u}^{k,i}\Bigg)
\end{aligned} \tag{S27}$$

The sum over time points:

$$\begin{aligned}
\mathcal{C} &= \sum_{t=1}^{T}\mathcal{B} \\
&= \sum_{i=1}^{M}\Bigg(\sum_{t=1}^{T}\Bigg[-\frac{1}{2}\alpha_i\sum_{j=1}^{M}(\alpha_{j,i}^{-1})^2\,\mathrm{tr}\left[\Psi_0^{j,i}(\boldsymbol{x}_{-t}^j) - \left(\boldsymbol{K}_{\boldsymbol{z}^{j,i},\boldsymbol{z}^{j,i}}^{j,i}\right)^{-1}\Psi_2^{j,i}(\boldsymbol{x}_{-t}^j)\right] \\
&- \log Z(\boldsymbol{I}\alpha_i) - \frac{1}{2}\alpha_i^{-1}\left[\mathrm{tr}(\boldsymbol{S}_{\boldsymbol{x}_t^i}) + \boldsymbol{\mu}_{\boldsymbol{x}_t^i}^T\boldsymbol{\mu}_{\boldsymbol{x}_t^i}\right]\Bigg] \\
&+ \sum_{j=1}^{M}\alpha_{j,i}^{-1}\left[\sum_{t=1}^{T}\boldsymbol{\mu}_{\boldsymbol{x}_t^i}^T\Psi_1^{j,i}(\boldsymbol{x}_{-t}^j)\right]\left(\boldsymbol{K}_{\boldsymbol{z}^{j,i},\boldsymbol{z}^{j,i}}^{j,i}\right)^{-1}\boldsymbol{u}^{j,i} \\
&- \frac{1}{2}\alpha_i\sum_{j=1}^{M}\sum_{k=1}^{M}\alpha_{j,i}^{-1}\alpha_{k,i}^{-1}\boldsymbol{u}^{j,iT}\left(\boldsymbol{K}_{\boldsymbol{z}^{j,i},\boldsymbol{z}^{j,i}}^{j,i}\right)^{-1}\left[\sum_{t=1}^{T}\Psi_2^{j,k,i}(\boldsymbol{x}_{-t}^j,\boldsymbol{x}_{-t}^k)\right]\left(\boldsymbol{K}_{\boldsymbol{z}^{k,i},\boldsymbol{z}^{k,i}}^{j,i}\right)^{-1}\boldsymbol{u}^{k,i}\Bigg)
\end{aligned} \tag{S28}$$

For every $i \in 1 \ldots M$ we may stack up the $\boldsymbol{u}^{:,i}$ into $\boldsymbol{u}^i$ and construct a large block matrices $\mathcal{F}^i$ and stacked matrices $\mathcal{G}^i$ with elements

$$\mathcal{F}_{j,k}^i = \alpha_i\alpha_{j,i}^{-1}\alpha_{k,i}^{-1}\left(\boldsymbol{K}_{\boldsymbol{z}^{j,i},\boldsymbol{z}^{j,i}}^{j,i}\right)^{-1}\left[\sum_{t=1}^{T}\Psi_2^{j,k,i}(\boldsymbol{x}_{-t}^j,\boldsymbol{x}_{-t}^k)\right]\left(\boldsymbol{K}_{\boldsymbol{z}^{k,i},\boldsymbol{z}^{k,i}}^{k,i}\right)^{-1} \tag{S29}$$

$$\mathcal{G}_j^i = \left(\alpha_{j,i}^{-1}\left[\sum_{t=1}^{T}\boldsymbol{\mu}_{\boldsymbol{x}_t^i}^T\Psi_1^{j,i}(\boldsymbol{x}_{-t}^j)\right]\left(\boldsymbol{K}_{\boldsymbol{z}^{j,i},\boldsymbol{z}^{j,i}}^{j,i}\right)^{-1}\right)^T \tag{S30}$$

For $j \neq k$: $\Psi_2^{j,k,i}(\boldsymbol{x}_{-t}^j, \boldsymbol{x}_{-t}^k) = \Psi_1^{j,i}(\boldsymbol{x}_{-t}^j)\Psi_1^{k,i}(\boldsymbol{x}_{-t}^k)$. Otherwise $\Psi_2^{j,j,i}(\boldsymbol{x}_{-t}^j, \boldsymbol{x}_{-t}^j) = \Psi_2^{j,i}(\boldsymbol{x}_{-t}^j)$.

The sum over time points must be expressed as a quadratic form w.r.t. the augmenting outputs $\boldsymbol{u}$:

$$\begin{aligned}
\mathcal{C} &= \sum_{i=1}^{M}\left[-\frac{1}{2}\boldsymbol{u}^{iT}\mathcal{F}^i\boldsymbol{u}^i + \boldsymbol{u}^{iT}\mathcal{G}^i + \mathcal{H}^i\right] \\
&= \sum_{i=1}^{M}\left[-\frac{1}{2}(\boldsymbol{u}^i - \mathcal{F}^{i-1}\mathcal{G}^i)^T\mathcal{F}^i(\boldsymbol{u}^i - \mathcal{F}^{i-1}\mathcal{G}^i) + \frac{1}{2}\mathcal{G}^{iT}\mathcal{F}^{i-1}\mathcal{G}^i + \mathcal{H}^i\right] \quad \text{(S31)} \\
\mathcal{C} &= \sum_{i=1}^{M}\mathcal{C}^i \quad \text{(S32)} \\
\mathcal{C}^i &= -\frac{1}{2}(\boldsymbol{u}^i - \mathcal{F}^{i-1}\mathcal{G}^i)^T\mathcal{F}^i(\boldsymbol{u}^i - \mathcal{F}^{i-1}\mathcal{G}^i) + \frac{1}{2}\mathcal{G}^{iT}\mathcal{F}^{i-1}\mathcal{G}^i + \mathcal{H}^i \quad \text{(S33)} \\
\mathcal{H}^i &= \sum_{t=1}^{T}\left[-\frac{1}{2}\alpha_i\sum_{j=1}^{M}(\alpha_{j,i}^{-1})^2\,\mathrm{tr}\left[\Psi_0^{j,i}(\boldsymbol{x}_{-t}^j) - \left(\boldsymbol{K}_{\boldsymbol{z}^{j,i},\boldsymbol{z}^{j,i}}\right)^{-1}\Psi_2^{j,i}(\boldsymbol{x}_{-t}^j)\right]\right. \\
&\quad \left. - \log Z(\boldsymbol{I}\alpha_i) - \frac{1}{2}\alpha_i^{-1}\left[\mathrm{tr}(\boldsymbol{S}_{\boldsymbol{x}_t^i}) + \boldsymbol{\mu}_{\boldsymbol{x}_t^i}^T\boldsymbol{\mu}_{\boldsymbol{x}_t^i}\right]\right] \quad \text{(S34)}
\end{aligned}$$

Finally, we may write the dynamics ELBO, also accounting for the optimal variational $q(\boldsymbol{u})$ (see eq. S8) and using $p(\boldsymbol{u}^i) = \prod_{j=1}^{M} p(\boldsymbol{u}^{ji}) = \prod_{j=1}^{M}\mathcal{N}(\boldsymbol{u}^{ji}|0, \boldsymbol{K}_{\boldsymbol{z}^{j,i},\boldsymbol{z}^{j,i}}) = \mathcal{N}(\boldsymbol{u}^i|0, \boldsymbol{K}_{\boldsymbol{z}^{:,i},\boldsymbol{z}^{:,i}})$ where $\boldsymbol{K}_{\boldsymbol{z}^{:,i},\boldsymbol{z}^{:,i}}$ is a block-diagonal covariance matrix:

$$\begin{aligned}
\mathcal{L}_{dyn}(\boldsymbol{\theta}) &\geq \log\int p(\boldsymbol{u})exp(\mathcal{C})d\boldsymbol{u} + H(q(\boldsymbol{x})) \\
&= \log\prod_{i=1}^{M}\int p(\boldsymbol{u}^i)exp(\mathcal{C}^i)d\boldsymbol{u}^i + H(q(\boldsymbol{x})) \\
&= \sum_{i=1}^{M}\left[\log\int p(\boldsymbol{u}^i)exp(-\frac{1}{2}(\boldsymbol{u}^i - \mathcal{F}^{i-1}\mathcal{G}^i)^T\mathcal{F}^i(\boldsymbol{u}^i - \mathcal{F}^{i-1}\mathcal{G}^i))d\boldsymbol{u}^i + \frac{1}{2}\mathcal{G}^{iT}\mathcal{F}^{i-1}\mathcal{G}^i + \mathcal{H}^i\right] + H(q(\boldsymbol{x})) \\
&= \sum_{i=1}^{M}\left[-\log Z(\mathcal{F}^{i-1} + \boldsymbol{K}_{\boldsymbol{z}^{:,i},\boldsymbol{z}^{:,i}}) - \frac{1}{2}\mathcal{G}^{iT}\mathcal{F}^{i-1}(\mathcal{F}^{i-1} + \boldsymbol{K}_{\boldsymbol{z}^{:,i},\boldsymbol{z}^{:,i}})^{-1}\mathcal{F}^{i-1}\mathcal{G}^i + \log Z(\mathcal{F}^{i-1})\right] \\
&\quad + \sum_{i=1}^{M}\left[\frac{1}{2}\mathcal{G}^{iT}\mathcal{F}^{i-1}\mathcal{G}^i + \mathcal{H}^i\right] + H(q(\boldsymbol{x})) \quad \text{(S35)}
\end{aligned}$$

This is even lower bound on the ELBO due to the sufficient statistics assumption for the $\boldsymbol{f}_t^{j,i}$ distribution. It is easy to notice that the optimal proposal $q(\boldsymbol{u}) = \prod_{i=1}^{M} q(\boldsymbol{u}^i)$ - factorized.

We optimize the full ELBO w.r.t. the parameters of $q(\boldsymbol{x})$, augmenting inputs $\boldsymbol{z}$, kernel parameters, and couplings $\boldsymbol{\alpha}$. After the optimization the optimal $q(\boldsymbol{u})$

is computed as:

$$
\begin{aligned}
q(\boldsymbol{u}^i) &= \frac{1}{Z} p(\boldsymbol{u}^i) exp(\mathcal{C}^i) \\
&= \frac{1}{Z} \mathcal{N}(\boldsymbol{u}^i | 0, \boldsymbol{K}_{\boldsymbol{z}^{:,i}, \boldsymbol{z}^{:,i}}) exp(-\frac{1}{2}(\boldsymbol{u}^i - \mathcal{F}^{i-1}\mathcal{G}^i)^T \mathcal{F}^i (\boldsymbol{u}^i - \mathcal{F}^{i-1}\mathcal{G}^i)) \\
&= \mathcal{N}(\boldsymbol{u}^i | (\boldsymbol{K}^{-1}_{\boldsymbol{z}^{:,i}, \boldsymbol{z}^{:,i}} + \mathcal{F}^i)^{-1}\mathcal{G}^i, (\boldsymbol{K}^{-1}_{\boldsymbol{z}^{:,i}, \boldsymbol{z}^{:,i}} + \mathcal{F}^i)^{-1})
\end{aligned} \tag{S36}
$$

3 ARD RBF kernel Ψ statistics. Full covariance variational parameters case.

Consider the following form of the approximate variational posterior distribution of $\boldsymbol{X}$:

$$
q(X) = \prod_{n=1}^{N} \mathcal{N}(\boldsymbol{x}_n | \boldsymbol{\mu}_n, S_n) \tag{S37}
$$

Here we derive the Ψ statistics for the variational lower bound for the case when the $\{S_n\}_{n=1\ldots N}$ are *full covariance matrices* and the ARD SE kernel is defined as:

$$
k(\boldsymbol{x}, \boldsymbol{x}') = \sigma_f^2 exp\left(-\frac{1}{2}\sum_{q=1}^{Q} \frac{(\boldsymbol{x}_q - \boldsymbol{x}'_q)^2}{\lambda_q}\right) \tag{S38}
$$

In matrix notation:

$$
\lambda = diag(\lambda_1 \ldots \lambda_Q) \tag{S39}
$$

$$
k(\boldsymbol{x}, \boldsymbol{x}') = \sigma_f^2 exp\left(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}')^T \lambda^{-1} (\boldsymbol{x} - \boldsymbol{x}')\right) \tag{S40}
$$

where λ_q are the ARD factors.

The Ψ_0 statistic is easy to calculate and it does not depend on the covariance matrix:

$$
\begin{aligned}
\Psi_0^n &= \int k(\boldsymbol{x}_n, \boldsymbol{x}_n) \mathcal{N}(\boldsymbol{x}_n | \boldsymbol{\mu}_n, S_n) d\boldsymbol{x}_n \\
&= \int \sigma_f^2 \mathcal{N}(\boldsymbol{x}_n | \boldsymbol{\mu}_n, S_n) d\boldsymbol{x}_n \\
&= \sigma_f^2 \int \mathcal{N}(\boldsymbol{x}_n | \boldsymbol{\mu}_n, S_n) d\boldsymbol{x}_n \\
&= \sigma_f^2
\end{aligned} \tag{S41}
$$

$$
\Psi_0 = \sum_{n=1}^{N} \Psi_0^n = N\sigma_f^2 \tag{S42}
$$

The Ψ_1 statistic:

$$
\begin{aligned}
(\Psi_1)_{nm} &= \int k(\boldsymbol{x}_n, \boldsymbol{z}_m)\mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_n, S_n)d\boldsymbol{x}_n \\
&= \int \sigma_f^2 exp\left(-\frac{1}{2}(\boldsymbol{x}_n - \boldsymbol{z}_m)^T\lambda^{-1}(\boldsymbol{x}_n - \boldsymbol{z}_m)\right)\mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_n, S_n)d\boldsymbol{x}_n \quad \text{(S43)}
\end{aligned}
$$

Next, complete the ARD SE kernel to a scaled Gaussian distribution:

$$
\begin{aligned}
(\Psi_1)_{nm} &= \sigma_f^2 \int \frac{Z(\lambda)}{Z(\lambda)}\sigma_f^2 exp\left(-\frac{1}{2}(\boldsymbol{x}_r - \boldsymbol{x}_m)^T\lambda^{-1}(\boldsymbol{x}_r - \boldsymbol{x}_m)\right)\mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_n, S_n)d\boldsymbol{x}_n \\
&= \sigma_f^2 Z(\lambda)\int \mathcal{N}(\boldsymbol{x}_n|\boldsymbol{z}_m, \lambda)\mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_n, S_n)d\boldsymbol{x}_n \quad \text{(S44)}
\end{aligned}
$$

Noticing the product of two Gaussians, which is an unnormalized Gaussian, the integral boils down to the scaling coefficient, which is a Gaussian value:

$$
\begin{aligned}
(\Psi_1)_{nm} &= \sigma_f^2 Z(\lambda)\mathcal{N}(\boldsymbol{z}_m|\boldsymbol{\mu}_n, \lambda + S_n) \\
&= \sigma_f^2 (2\pi)^{Q/2}\sqrt{|\lambda|}\frac{1}{(2\pi)^{Q/2}\sqrt{|\lambda + S_n|}} exp\left(-\frac{1}{2}(\boldsymbol{z}_m - \boldsymbol{\mu}_n)^T(\lambda + S_n)^{-1}(\boldsymbol{z}_m - \boldsymbol{\mu}_n)\right) \\
&= \sigma_f^2 \frac{\sqrt{|\lambda|}}{\sqrt{|\lambda + S_n|}} exp\left(-\frac{1}{2}(\boldsymbol{z}_m - \boldsymbol{\mu}_n)^T(\lambda + S_n)^{-1}(\boldsymbol{z}_m - \boldsymbol{\mu}_n)\right) \\
&= \sigma_f^2 \sqrt{\frac{\prod_{q=1}^{Q}\lambda_q}{|\lambda + S_n|}} exp\left(-\frac{1}{2}(\boldsymbol{z}_m - \boldsymbol{\mu}_n)^T(\lambda + S_n)^{-1}(\boldsymbol{z}_m - \boldsymbol{\mu}_n)\right) \quad \text{(S45)} \\
& \quad \text{(S46)}
\end{aligned}
$$

The Ψ_2 statistic integral can be solved in the same manner:

$$
\begin{aligned}
(\Psi_2)_{mm'} &= \int k(\boldsymbol{x}_n, \boldsymbol{z}_m)k(\boldsymbol{z}_{m'}, \boldsymbol{x}_n)\mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_n, S_n)d\boldsymbol{x}_n \\
&= (\sigma_f^2 Z(\lambda))^2 \int \mathcal{N}(\boldsymbol{x}_n|\boldsymbol{z}_m, \lambda)\mathcal{N}(\boldsymbol{x}_n|\boldsymbol{z}_{m'}, \lambda)\mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_n, S_n)d\boldsymbol{x}_n \quad \text{(S47)}
\end{aligned}
$$

Here we have to multiply the Gaussians twice:

$$
\begin{aligned}
(\Psi_2)_{mm'} &= (\sigma_f^2 Z(\lambda))^2 \int \mathcal{N}(\boldsymbol{z}_{m'}|\boldsymbol{z}_m, 2\lambda) \\
&\times \mathcal{N}(\boldsymbol{x}_n|(\lambda^{-1}+\lambda^{-1})^{-1}(\lambda^{-1}\boldsymbol{z}_m + \lambda^{-1}\boldsymbol{z}_{m'}), (\lambda^{-1}+\lambda^{-1})^{-1})\mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_n, S_n)d\boldsymbol{x}_n \\
&= (\sigma_f^2 Z(\lambda))^2 \int \mathcal{N}(\boldsymbol{z}_{m'}|\boldsymbol{z}_m, 2\lambda)\mathcal{N}(\boldsymbol{x}_n|\frac{\boldsymbol{z}_m+\boldsymbol{z}_{m'}}{2}, \frac{\lambda}{2})\mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_n, S_n)d\boldsymbol{x}_n \\
&= (\sigma_f^2 Z(\lambda))^2 \mathcal{N}(\boldsymbol{z}_{m'}|\boldsymbol{z}_m, 2\lambda) \int \mathcal{N}(\boldsymbol{x}_n|\frac{\boldsymbol{z}_m+\boldsymbol{z}_{m'}}{2}, \frac{\lambda}{2})\mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_n, S_n)d\boldsymbol{x}_n \\
&= (\sigma_f^2 Z(\lambda))^2 \mathcal{N}(\boldsymbol{z}_{m'}|\boldsymbol{z}_m, 2\lambda) \int \mathcal{N}(\boldsymbol{\mu}_n|\frac{\boldsymbol{z}_m+\boldsymbol{z}_{m'}}{2}, \frac{\lambda}{2}+S_n)\mathcal{N}(\boldsymbol{x}_n|mean, precision)d\boldsymbol{x}_n \\
&= (\sigma_f^2 Z(\lambda))^2 \mathcal{N}(\boldsymbol{z}_{m'}|\boldsymbol{z}_m, 2\lambda)\mathcal{N}(\boldsymbol{\mu}_n|\frac{\boldsymbol{z}_m+\boldsymbol{z}_{m'}}{2}, \frac{\lambda}{2}+S_n) \int \mathcal{N}(\boldsymbol{x}_n|mean, precision)d\boldsymbol{x}_n \\
&= (\sigma_f^2 Z(\lambda))^2 \mathcal{N}(\boldsymbol{z}_{m'}|\boldsymbol{z}_m, 2\lambda)\mathcal{N}(\boldsymbol{\mu}_n|\frac{\boldsymbol{z}_m+\boldsymbol{z}_{m'}}{2}, \frac{\lambda}{2}+S_n) \\
&= \sigma_f^4 (2\pi)^Q (\prod_{q=1}^{Q} \lambda_q)\mathcal{N}(\boldsymbol{z}_{m'}|\boldsymbol{z}_m, 2\lambda)\mathcal{N}(\boldsymbol{\mu}_n|\frac{\boldsymbol{z}_m+\boldsymbol{z}_{m'}}{2}, \frac{\lambda}{2}+S_n) \qquad \text{(S48)}
\end{aligned}
$$

For the case of the diagonal covariance S_n the Ψ_3 statistic looks simpler [2] as it does not require inversion of the $\lambda + S_n$ and $\frac{\lambda}{2} + S_n$ matrices.

4 Relationship between ELBO and MSE

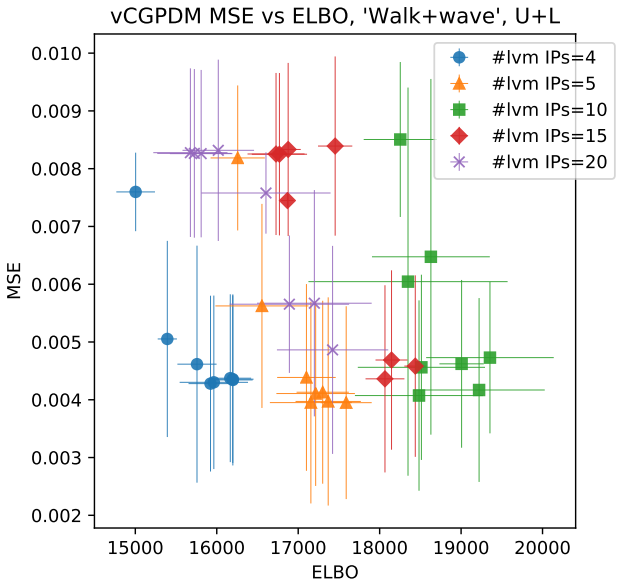


Figure 1: Mean-squared kinematics error, MSE, vs. evidence lower bound, ELBO, for different number of LVM IPs, indicated by symbols. The negative correlation between ELBO and MSE for a given number of LVM IPs is clearly visible. Furthermore, note that the highest ELBO corresponds to an MSE that is very close to the optimal (lowest) one, i.e. ELBO can be used for model selection.

5 Learning time of vCGPDM and MAP CGPDM

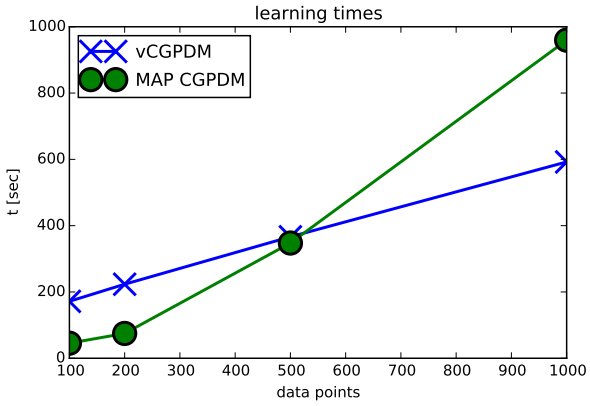


Figure 2: Learning times, including Theano compile times, for a three-part vCGPDM with 10 IPs for all parts and corresponding MAP CGPDM. The linear learning time scaling of the vCGPDM is evident, whereas the MAP CGPDM shows cubic scaling. Thus, the vCGPDM can be used on large data sets which is infeasible for the MAP CGPDM.

References

[1] Michalis K. Titsias. Variational learning of inducing variables in sparse Gaussian processes. In David A. Van Dyk and Max Welling, editors, *AISTATS*, volume 5 of *JMLR Proceedings*, pages 567–574. JMLR.org, 2009.

[2] Michalis K. Titsias and Neil D. Lawrence. Bayesian Gaussian process latent variable model. In *Proc. 13th AISTATS*, pages 844–851, 2010.

entropy

MDPI

Article

Making the Coupled Gaussian Process Dynamical Model Modular and Scalable with Variational Approximations †

Dmytro Velychko *, Benjamin Knopp and Dominik Endres *

Department of Psychology, University of Marburg, Gutenbergstr. 18, 35032 Marburg, Germany; benjamin.knopp@uni-marburg.de

* Correspondence: dmytro.velychko@uni-marburg.de (D.V.); dominik.endres@uni-marburg.de (D.E.)

† This paper is an extended version of our paper published in the 26th International Conference on Artificial Neural Networks (ICANN 2017), Alghero, Italy, 11–14 September, 2017.

Received: 27 July 2018; Accepted: 20 September 2018; Published: 21 September 2018

Abstract: We describe a sparse, variational posterior approximation to the Coupled Gaussian Process Dynamical Model (CGPDM), which is a latent space coupled dynamical model in discrete time. The purpose of the approximation is threefold: first, to reduce training time of the model; second, to enable modular re-use of learned dynamics; and, third, to store these learned dynamics compactly. Our target applications here are human movement primitive (MP) models, where an MP is a reusable spatiotemporal component, or "module" of a human full-body movement. Besides re-usability of learned MPs, compactness is crucial, to allow for the storage of a large library of movements. We first derive the variational approximation, illustrate it on toy data, test its predictions against a range of other MP models and finally compare movements produced by the model against human perceptual expectations. We show that the variational CGPDM outperforms several other MP models on movement trajectory prediction. Furthermore, human observers find its movements nearly indistinguishable from replays of natural movement recordings for a very compact parameterization of the approximation.

Keywords: Gaussian processes; variational methods; movement primitives; modularity

1. Introduction

Two formidable problems that the human brain has to solve are planning and execution of movements of its body. As a means to simplify these problems while keeping a sufficient degree of control flexibility for a wide range of tasks, modular movement primitives (MP) have been suggested (see [1,2] for reviews). There is no universally accepted definition of the term "movement primitive". For the purposes of this paper, an MP is a spatiotemporal component of a human (full-body) movement that may be produced by mapping a latent state onto observable variables, such as joint angles. The latent state can be generated by dynamical systems [3] or source functions [4,5]. "Modular" usually refers to the existence of an operation which allows for the spatial, temporal or spatiotemporal combination of (simple) primitives into (complex) movements.

Two prominent examples, where this operation is the linear combination of stereotypical time-courses or muscle-coactivations, are called temporal MP-models [6–9] or spatial MP-models [10,11]. While these models are inherently modular, the assumption of stereotyped MPs makes it difficult for a control system built out of these primitives to respond to perturbations. A type of MP which can be controlled on-line more easily is the dynamical MP (DMP) [3], which has been developed for robotics applications. In this approach, each primitive is encoded by a canonical second order differential

equation with guaranteeable stability properties and learnable parameters. A DMP can generate both discrete (e.g., reaching) and rhythmic (e.g., walking) movements and drives the trajectory of one degree of freedom, e.g., a joint angle. Modularity arises because of the latter property, which might be viewed as an "extreme" form of the modularization that we investigate here, where one movement module might affect several degrees of freedom. Similarly, recent extensions of the DMP framework allow for the reuse of a DMP across end-effectors via kinematical mappings [12] or across tasks [13].

We describe a model that learns MPs composed of coupled dynamical systems and associated kinematics mappings, where both components are learned, thus lifting the DMP's restriction of canonical dynamics. We build on the Coupled Gaussian Process Dynamical Model (CGPDM) by [14], which combines the advantages of modularity and flexibility in the dynamics, at least theoretically. In a CGPDM, the temporal evolution functions for the latent dynamical systems are drawn out of a Gaussian process (GP) prior [15]. These dynamical systems are then coupled probabilistically, and the result is mapped onto observations by functions drawn from another GP. One drawback of the CGPDM is its fully non-parametric nature, which results in cubic scaling (with the dataset size) of learning complexity and quadratic scaling of MP storage size, i.e., the CGPDM can not be learned from large data sets, and its effective parameterization is not compact. We improve both scalability and compactness with deterministic, sparse variational approximations [16]. In this sparse variational CGPDM, each MP is parameterized by a small set of inducing points (IPs) and associated inducing values (IVs), leading to a compact representation with linear scaling of the training complexity in the number of data points, and constant storage requirements. This compactness is important for real-world applicability of the model, since there might be more primitives than muscles (or actuators) across tasks, as pointed out by Bizzi and Cheung [17]: the motor "code" might be sparse and overcomplete, similar to the sparse codes in early vision [18]. Table 1 provides an overview of the key MP models which we compare in this paper.

Table 1. Overview of movement primitive models compared in this paper. (v)CGPDM, (variational) coupled Gaussian process dynamical model; (v)GPDM, (variational) Gaussian process dynamical model; TMP, temporal movement primitives; DMP, dynamical movement primitives. Modular, learns reusable MPs. Scalable, below cubic learning complexity with respect to the data set size; Compact, size of the effective parameterization does not grow with the data set size; Canonical dynamics, dynamics model specified before learning; Learned dynamics, dynamics model is a free-form function.

	Modular	**Scalable**	**Compact**	**Canonical Dynamics**	**Learned Dynamics**
vCGPDM	✓	✓	✓	x	✓
CGPDM	x	x	x	x	✓
vGPDM	x	✓	✓	x	✓
GPDM	x	x	x	x	✓
TMP	✓	✓	✓	x	x
DMP	✓	✓	✓	✓	x

Our target application here is human movement modeling, but the vCGPDM could be easily applied to other systems where modularized control is beneficial, e.g., humanoid robotics [9].

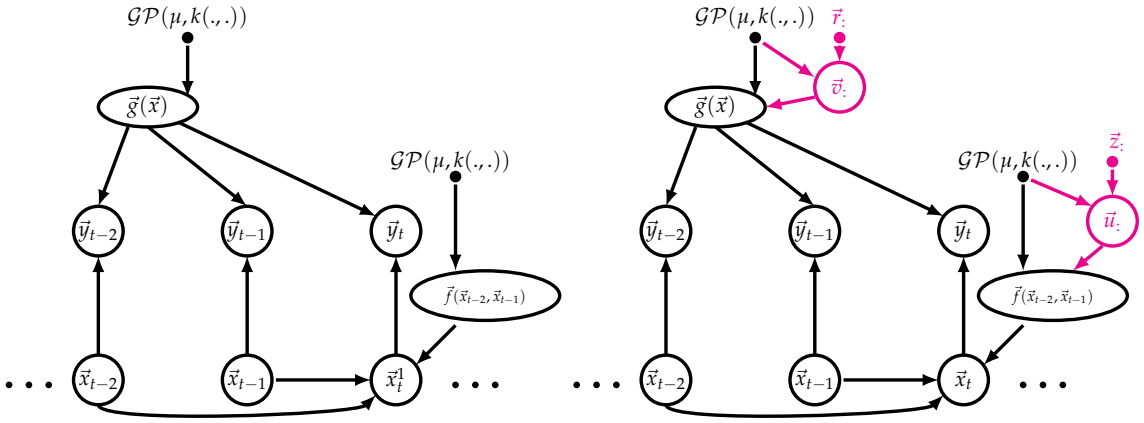
We briefly review related work in Section 2 and introduce the vCGPDM in Section 3. The derivation of the variational approximation is outlined in Section 4. In Section 5, we first illustrate the vCGPDM on artificial data. Second, we benchmark the vCGPDM against other MP models. Third, we perform an experiment to quantify the degree of of human-tolerable sparseness in a psychophysics experiment. Fourth, we demonstrate modular movement composition with the vCGPDM. In Section 6, we propose future research directions based on our work.

This paper is a substantially extended version of our earlier conference publication [19].

2. Related Work

The Gaussian process (GP) is a machine learning staple for classification and regression tasks [15]. A GP is a prior on functions $\mathbb{R}^Q \rightarrow \mathbb{R}$ from a Q-dimensional input space to one-dimensional output. By drawing D times from the GP, functions from $\mathbb{R}^Q \rightarrow \mathbb{R}^D$ can be realized. Its advantages include theoretical elegance, tractability and closed-form solutions for posterior densities. Its main disadvantage is cubic runtime scaling with the number of data points. Several solutions have been proposed for this problem. Many of these involve a sparse representation of the posterior process via a small set of IPs, which may [20] or may not be a subset of the data points [21]. If the input space is unobserved, one obtains a GP latent variable model (GPLVM), for which sparse approximations have also been devised [22]. One problem with sparse GP approximations is their tendency to overfit [22], leading to incorrect variance predictions [23]. In that paper, it is also demonstrated that the problem can be alleviated by a variational approximation, which prompted us to develop a similar approach for the CGPDM: as in [24], we extend the sparse GPLVM in time, but we use an autoregressive dynamical system.

If the temporal evolution function of this dynamical system is also drawn from a GP, the resulting model is called Gaussian Process Dynamical Model (GPDM), which can be learned by maximum-a-posteriori approximation if the observed dimension D is greater than the latent dimension Q [25]. Figure 1 (left) shows a graphical model representation of the GPDM, and introduces the related notation which we use throughout the paper. Slices ":" indicate collections of variables along one integer index. Multiple slices refer to collections along multiple indices, e.g., $\vec{x}_{::}$ are the latent variables of all parts and time-steps. The GPDM can model the variability of human movements and has been used for computer animation with style control [26–28]. It has also been used with an additional switching prior on the dynamics for motion tracking and recognition [29] and deep variants have been devised [30]. However, with the exception of the coupled GPDM [14,19], all these approaches have a "monolithic" latent space and thus lack the modularity of MPs. One reason for this might be the fact that, for the maximum-a-posteriori approximation to work, the latent space has to be lower-dimensional than the observed space, $Q \ll D$. If, as explained above, we want a modular, possibly overcomplete (i.e., the effective $Q > D$) set of MPs, we need learning approaches that are robust to overfitting. The works of Frigola et al. [31] indicate that such approaches may be obtained with variational approximations. In the following, we therefore introduce a variational approximation to CGPDM learning and inference based on an approach similar to Frigola et al. [32], but, as in [30], we aim to obviate the need for sampling altogether to allow for fast, repeatable trajectory generation. While deriving a variational approximation is not trivial, we expect it to avoid overfitting and yield a good bound on the marginal likelihood [33]. Figure 1 (right) shows the graphical model of the GPDM augmented by IPs and IVs. This augmentation yields a tractable variational approximation to the GPDM's posterior [30].



Notation and Abbreviations (v)GPDM

Gaussian process dynamical model	GPDM	variational approximation to posterior of GPDM	vGPDM
discrete time index	$t = 1, \ldots, T$	mean and kernel function	$\mu,\ k(.,.)$
latent space dimensionality	Q	latent states	$\vec{x}_t \in \mathbb{R}^Q$
observed space dimensionality	D	observable variables	$\vec{y}_t \in \mathbb{R}^D$
Gaussian process prior	$\mathcal{GP}(\mu, k(.,.))$	Inducing points/values	IP/IV
latent-to-observed function	$\vec{g}(\vec{x}_t)$	dynamics function	$\vec{f}(\vec{x}_{t-2}, \vec{x}_{t-1})$
IP of latent-to-observed function	$\vec{r}_: = (\vec{r}_1, \vec{r}_2, \ldots)$	IP of dynamics function	$\vec{z}_: = (\vec{z}_1, \vec{z}_2, \ldots)$
IV of latent-to-observed function	$\vec{v}_: = (\vec{v}_1, \vec{v}_2, \ldots)$	IV of dynamics function	$\vec{u}_: = (\vec{u}_1, \vec{u}_2, \ldots)$

Figure 1. Modular building blocks of the vCGPDM. (**Left**) The Gaussian process dynamical model (GPDM). A latent, second order dynamics model generates a time-series of vector-valued random variables $\vec{x}_t$ which are drawn from a multivariate Gaussian distribution with mean function $\vec{f}(\vec{x}_{t-2}, \vec{x}_{t-1})$. The components of this mean function are drawn from a Gaussian Process $\mathcal{GP}(\mu, k(.,.))$. Each observable $\vec{y}_t$ is drawn from multivariate Gaussian distribution with mean function $\vec{g}(\vec{x}_t)$, which have a Gaussian process prior, too. (**Right**) The GPDM augmented with inducing points and values for a sparse representation of the posterior process [23]. This enables faster variational Bayesian learning and inference, because the augmented GPs are effectively parameterized by these points (here, $\vec{r}_:, \vec{z}_:$) and corresponding values (here, $\vec{v}_:, \vec{u}_:$) rather than by the full dataset. They may be thought of as prototypical examples of the corresponding functions, e.g., $\vec{v}_k = \vec{g}(\vec{r}_k)$. Slice notation ":" indicates collections of variables. For details, see text.

3. The Model

The basic building blocks, or "parts", of the CGPDM are a number of GPDMs run in parallel. In the context of human movement modeling, e.g., they may be thought of as body parts. A part evolves in discrete time $t = 0, \ldots, T$ and is endowed with a Q-dimensional latent space, a D-dimensional observed space and second-order autoregressive dynamics described by a function $\vec{f}(\vec{x}_{t-2}, \vec{x}_{t-1})$. The component functions $\left(\vec{f}(\vec{x}_{t-2}, \vec{x}_{t-1})\right)_q$ have a Gaussian process prior $\mathcal{GP}(\mu, k(.,.))$ with mean function μ and kernel $k(.,.)$ Second-order dynamics seem to be a good choice for our target application of human movement modeling [34], but the order can be easily altered simply by concatenating previous states into one larger vector. Let $\vec{x}_t \in \mathbb{R}^Q$ the state of latent space of the part at time t (see Figure 1, left). This latent state produces observations $\vec{y}_t \in \mathbb{R}^D$ via the function $\vec{g}(\vec{x}_t)$ as well as isotropic Gaussian noise with variance β. The components $(\vec{g}(\vec{x}_t))_d$ of this function are drawn from a Gaussian process prior, too. GPDMs can be learned from data via a combination of exact marginalization and maximum-a-posteriori learning of the latent dynamics [25].

While the GPDM is a very expressive model, it suffers from poor runtime and memory scaling with the data set size due to its non-parametric nature, which it inherits from the involved GPs. We remedied this problem by an approach pioneered in [23]: augmenting the GPs with inducing points (IPs, here: $\vec{r}_:, \vec{z}_:$) and associated inducing values (IVs, $\vec{v}$, $\vec{u}$) (see Figure 1, right). These IP/IV pairs might

be thought of as prototypical examples of the mappings represented by the corresponding functions, e.g., $\vec{v}_k^i = \vec{g}^i(\vec{r}_k^i)$. Note that the IPs are not drawn from a prior, whereas the IVs are. Hence, the latter are model parameters, whereas the former are not: IPs are merely parameters of the approximation, or "variational parameters". This augmentation allows for the derivation of a closed-form evidence lower bound (ELBO) on the marginal likelihood of the model.

In a CGPDM, the latent spaces of the parts are coupled to each other. We index parts by superscripts $i = 1, \ldots, M$. The index notation in such models can be confusing for first-time readers, we provide a notation and index overview in the tables below Figure 2. In our target application, the coupling may reflect the influences which parts of an articulated body have on each other during the execution of a movement. The coupling is implemented by having the parts make Gaussian-distributed predictions $\vec{x}_t^{i,j}$ about each other's latent states with means generated by $M \times M$ many mean coupling functions $\vec{f}^{i,j}(\vec{x}_{t-2}^i, \vec{x}_{t-1}^i)$ and coupling variances $\alpha^{i,j}$. i indexes the origin part of a coupling, and j its target. Thus, $\vec{f}^{i,i}$ refers to the dynamics function for part i. The components of $\vec{f}^{i,j}(\vec{x}_{t-2}^i, \vec{x}_{t-1}^i)$ are drawn from GPs. As described in [14], these predictions are combined with a product-of-experts construction (PoE [35]), including the predictions which a part makes about its own future. A product-of-experts construction forces the experts to agree on one prediction (Equation (1), left), which amounts to multiplying the individual predictions (Equation (1), right) and renormalizing (Equation (1), middle):

$$p(\vec{x}_t^j | \vec{f}^{:,j}(\vec{x}_{t-2}^i, \vec{x}_{t-1}^j), \alpha^{:,j}) = \frac{\exp\left[-\frac{1}{2\alpha^j}\left(\vec{x}_t^j - \alpha^j \sum_i \frac{\vec{f}^{i,j}(\vec{x}_{t-2}^i, \vec{x}_{t-1}^j)}{\alpha^{i,j}}\right)^2\right]}{(2\pi\alpha^j)^{\frac{Q^j}{2}}} \propto \prod_i \mathcal{N}\left(\vec{x}_t^j | \vec{f}^{i,j}(\vec{x}_{t-2}^i, \vec{x}_{t-1}^i), \alpha^{i,j}\right) \quad (1)$$

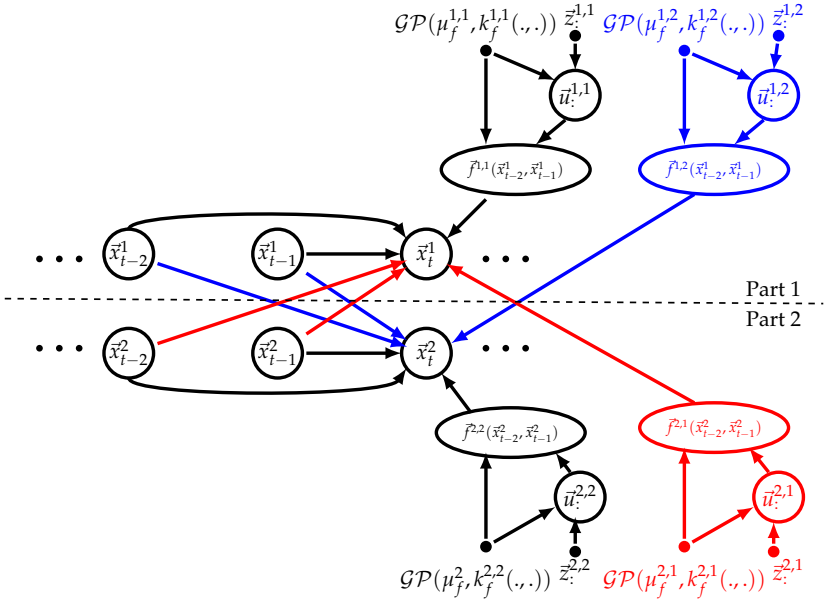
where $\alpha^j = \left(\sum_i (\alpha^{i,j})^{-1}\right)^{-1}$.

To understand the function of the $\alpha^{i,j}$, consider the form of Equation (1): the smaller a given variance, the more important the prediction of the generating part. We optimize the $\alpha^{i,j}$ during learning, letting the model discover which couplings are important for predicting the data. In other words, whenever an $\alpha^{i,j}$ is small compared to $\alpha^{i' \neq i,j}$, then part i is able to make a prediction about part j with (relatively) high certainty. Furthermore, the $\alpha^{i,j}$ can be modulated after learning to generate new movements, as shown below.

In the following, we denote all relevant timesteps before time t with subscript $-t$, e.g., $\vec{x}_{-t}^j = (\vec{x}_{t-2}^j, \vec{x}_{t-1}^j)$ for a second-order dynamics model. We showed in [14] that the individual predictions of part i about part j, $\vec{x}_t^{i,j}$ can be exactly marginalized, leading to a GPDM-like model for each part with a dynamics kernel given by the $\alpha_{i,j}$-weighted mean of the individual coupling kernels:

$$k_f^j\left(\vec{x}_{-t}^1, \vec{x}_{-t}^{1\prime}, \ldots, \vec{x}_{-t}^M, \vec{x}_{-t}^{M\prime}\right) = \alpha^{j2} \sum_{i=1}^{M} \frac{k_f^{i,j}(\vec{x}_{-t}^i, \vec{x}_{-t}^{i\prime}, \vec{x}_{-t}^j, \vec{x}_{-t}^{j\prime})}{\alpha^{i,j2}} \quad (2)$$

However, doing so results in a model which lacks modularity: after learning, it is difficult to separate the parts from each other, and recombine them for the production of new movements that were not in the training data. We facilitate this modular recombination by restating CGPDM learning such that we can keep an explicit, sparse representation of the coupling functions. Another reason for a sparse representation is that the CGPDM exhibits cubic run time scaling with the data points, which it inherits from the composing GPDMs. To remedy these problems, we follow the treatment in [16,30]: we augment the model with IPs $\vec{r}_:^i$ and associated IVs $\vec{v}_:^i$ such that $\vec{g}^i(\vec{r}_k^i) = \vec{v}_k^i$ for the latent-to-observed mappings $g^i()$. Then, we condition the probability density of the function values of $g^i()$ on these IPs/IVs, which we assume to be a sufficient statistic. Likewise, we reduce the computational effort for learning the dynamics and coupling mappings by inducing them through $\vec{z}_:^{i,j}$ and $\vec{u}_:^{i,j}$ (also known as "dynamics IPs/IVs"). See Figure 2 for a graphical representation of the augmented model.



Notation and Abbreviations (v)CGPDM

Coupled Gaussian process dynamical model	CGPDM	variational approximation to posterior of CGPDM	vCGPDM
latent states, part i at time t	$\vec{x}_t^i \in \mathbb{R}^Q$	Inducing points/values	IP/IV
latent states, part i before t	$\vec{x}_{-t}^i$	latent predictions of part j about part i at time t	$\vec{x}_t^{j,i}$
observable variables, part i at time t	$\vec{y}_t^i \in \mathbb{R}^D$	Gaussian process prior	$\mathcal{GP}(\mu, k(.,.))$
dynamics function, part i	$\vec{f}^{i,i}(\vec{x}_{t-2}^i, \vec{x}_{t-1}^i)$	latent-to-observed function, part i	$\vec{g}^i(\vec{x}_t^i)$
coupling function, part j-to-i	$\vec{f}^{j,i}(\vec{x}_{t-2}^j, \vec{x}_{t-1}^j)$	coupling variance, part j-to-i	$\alpha^{j,i}$
mean and kernel of $\mathcal{GP}$ prior on dynamics function, part i	$\mu_f^i,\ k_f^{i,i}(.,.)$	mean and kernel of $\mathcal{GP}$ prior on latent-to-observed function, part i	$\mu_g^i,\ k_g^{i,i}(.,.)$
IP of latent-to-observed function, part i	$\vec{r}_:^i = (\vec{r}_1^i, \vec{r}_2^i, \ldots)$	IP of dynamics function, part j-to-i	$\vec{z}_:^{j,i} = (\vec{z}_1^{j,i}, \vec{z}_2^{j,i}, \ldots)$
IV of latent-to-observed function, part i	$\vec{v}_:^i = (\vec{r}_1^i, \vec{r}_2^i, \ldots)$	IV of dynamics function, part j-to-i	$\vec{u}_:^{j,i} = (\vec{r}_1^{j,i}, \vec{r}_2^{j,i}, \ldots)$

Index Summary

$\vec{x}_{time}^{part}$	$\vec{u}_{IV-index}^{from-part,to-part}$	$\vec{f}_{time}^{from-part,to-part}$	$\vec{z}_{IP-index}^{from-part,to-part}$
$\vec{g}_{time}^{part}$	$\vec{v}_{IV-index}^{part}$	$\vec{r}_{IP-index}^{part}$	

d-th component of vector $\vec{z}_3^{2,1}$: $(\vec{z}_3^{2,1})_d$

Figure 2. (**Top**): Graphical model representation of the augmented Coupled Gaussian Process Dynamical Model (vCGPDM). Shown is a model with two parts, indicated by the superscripts $i, j \in \{1, 2\}$. Each part is a vGPDM (see Figure 1), augmented with inducing points $\vec{z}_:^{i,j}$ and values $\vec{u}_:^{i,j}$ for variational inference and learning, and modular re-composition of learned GPDM components. Observed variables $\vec{y}_t^i$ and latent-to-observed mappings $\vec{g}^i(\vec{x}_t^i)$ omitted for clarity. The vGPDMs interact by making predictions about each other's latent space evolution via functions $\vec{f}^{i,j}(\vec{x}_{t-2}^i, \vec{x}_{t-1}^i)$, here $\vec{f}^{1,2}()$ and $\vec{f}^{2,1}()$. Their predictions are product-of-experts combined with the predictions made by each GPDM's dynamics model (functions $\vec{f}^{i,i}(\vec{x}_{t-2}^i, \vec{x}_{t-1}^i)$). (**Bottom**): Notation and index summaries.

Besides introducing IPs, computing an ELBO requires a simplifying assumption about the latent state posterior, which is intractable. We choose a posterior distribution q over the latent states $\vec{x}_t^i$ that factorizes across time steps $0, \ldots, T$, parts $1, \ldots, M$ and latent dimensions $1, \ldots, Q^i$ within parts. Furthermore, we assume that the individual distributions are Gaussian:

$$q(\vec{x}_0^1, \ldots, \vec{x}_T^M) = \prod_{t=0}^{T} \prod_{i=1}^{M} \prod_{q=1}^{Q^i} q((\vec{x}_t^i)_q) \; ; \;\; q((\vec{x}_t^i)_q) = \mathcal{N}(\mu_{t,q}^i, \sigma_{t,q}^{2,i}). \tag{3}$$

While this approximation is clearly a gross simplification of the correct latent state posterior, with the right choice of kernels, an ELBO can now be computed. Our approximation assumption (Equation (3)) seems appropriate for human movement data, see Section 5. Whether it is also useful for other data remains to be determined.

As for a tractable kernel, we decided to use an ARD (automatic relevance determination) squared exponential kernel [36] for every part-i-to-j prediction GP:

$$k^{i,j}(\vec{x}_{-t}^i, \vec{x}_{-t}^{i\,\prime}) = \exp\left(-\frac{1}{2} \sum_{t \in -t} \sum_{q}^{Q^i} \frac{((\vec{x}_t^i)_q - (\vec{x}_t^{i\,\prime})_q)^2}{\lambda_q^{i,j,t}} \right). \tag{4}$$

and a radial basis function kernel for the latent-to-observed mappings. Next, we outline the key steps of the derivation of the ELBO.

4. Computing an Evidence Lower Bound for the vCGPDM: An Overview

In this section, we provide an overview of the derivation of the evidence lower bound (ELBO) for the vCGPDM; for details, the reader is referred to Appendix C. We construct a sparse variational approximation by augmenting each of the $M \times M$ dynamics and coupling mappings $\vec{f}^{i,j}()$ with IPs and IVs. The variational distribution of the latent variables, $q(\vec{x}_{:}^{:}) = q(\vec{x}_1^1, \ldots, \vec{x}_T^M)$ factorizes according to Equation (3). We let $q(\vec{u}_{:}^i)$ and $q(\vec{v}_{:}^i)$ be unconstrained distributions, which will turn out to be multivariate Gaussians. In the following, we denote the coupling function values at t with $\vec{f}_t^{i,j} = f^{i,j}(\vec{x}_{-t}^i)$ and likewise $\vec{g}_t^i = g^i(\vec{x}_t^i)$. The factor structure of the joint density of the augmented model follows from the graphical model (see Figures 1 and 2):

$$p(\vec{y}_{:}^{:}, \vec{g}_{:}^{:}, \vec{v}_{:}^{:}, \vec{x}_{:}^{:}, \vec{f}_{:}^{:,:}, \vec{u}_{:}^{:,:} | \vec{z}_{:}^{:,:}, \vec{r}_{:}^{:}) = p(\vec{y}_{:}^{:} | \vec{g}_{:}^{:}) p(\vec{g}_{:}^{:} | \vec{x}_{:}^{:}, \vec{v}_{:}^{:}, \vec{r}_{:}^{:}) p(\vec{v}_{:}^{:} | \vec{r}_{:}^{:}) p(\vec{x}_{:}^{:}, \vec{f}_{:}^{:,:} | \vec{u}_{:}^{:,:}, \vec{z}_{:}^{:,:}) p(\vec{u}_{:}^{:,:} | \vec{z}_{:}^{:,:}) \tag{5}$$

Note that we marginalized (most of) the functions $f^{:,:}()$ here, keeping only their values at the latent points $\vec{x}_{:}^{:}$ and at the IPs. Hence the dependence of $\vec{g}_{:}^{:}$ on $\vec{r}_{:}^{:}$. Likewise, $\vec{f}_{:}^{:,:}$ depends on $\vec{z}_{:}^{:,:}$. For easier notation, we omit spelling out the dependence of the IVs on the IPs in the following. Thus,

$$p(\vec{y}_{:}^{:} | \vec{g}_{:}^{:}) = \prod_{i=1}^{M} \prod_{d=1}^{D_i} p((\vec{y}_{:}^{i})_d | (\vec{g}_{:}^{i})_d) \tag{6}$$

$$p(\vec{g}_{:}^{:} | \vec{x}_{:}^{:}, \vec{v}_{:}^{:}) = \prod_{i=1}^{M} \prod_{d=1}^{D_i} p((\vec{g}_{:}^{i})_d | \vec{x}_{:}^{i}, (\vec{v}_{:}^{i})_d) \tag{7}$$

$$p(\vec{v}_{:}^{:}) = \prod_{i=1}^{M} p(\vec{v}_{:}^{i}); \;\; p(\vec{u}_{:}^{:,:}) = \prod_{i=1}^{M} \prod_{j=1}^{M} p(\vec{u}_{:}^{j,i}). \tag{8}$$

where Equation (6) follows from the assumption of independent observation noise. Equation (7) is a consequence of the Gaussian process prior on the $g^i()$, which makes the components of $\vec{g}_t^i$ independent. The density of the latent variables and the individual parts' predictions can be factorized as:

$$p(\vec{x}_{:}^{:}, \vec{f}_{:}^{:,:} | \vec{u}_{:}^{:,:}) = p(\vec{x}_{:}^{:} | \vec{f}^{:,:}, \vec{u}_{:}^{:,:}) p(\vec{f}_{:}^{:,:} | \vec{u}_{:}^{:,:}) \tag{9}$$

with

$$p(\vec{x}_{:}^{:}|\vec{f}_{:}^{:,:},\vec{u}_{:}^{:,:}) = \prod_{t=2}^{T}\prod_{i=1}^{M} p(\vec{x}_t^i|\vec{f}_t^{:,i},\alpha^{:,i}) \tag{10}$$

$$p(\vec{f}_{:}^{:,:}|\vec{u}_{:}^{:,:}) = \prod_{t=2}^{T}\prod_{i=1}^{M}\prod_{j=1}^{M} p(\vec{f}_t^{j,i}|\vec{f}_{1:t-1}^{j,i},\vec{x}_{0:t-1}^{j},\vec{u}_{:}^{j,i}) \tag{11}$$

where Equation (10) follows from the graphical model and the product-of-experts construction (Equation (1)). An empty slice ($t < 2$ for a second-order dynamics model) implies no conditioning. The first two latent states at $t = 0, 1$ are drawn from independent Gaussians, $\prod_{i=1}^{M} p(\vec{x}_0^i)p(\vec{x}_1^i)$. Equation (11) is one possible way of factorizing the augmented Gaussian process prior on the coupling function values: when $\vec{f}^{i,j}()$ is marginalized, the function values at time t depend on all past function values and latent states. We use this particular factorization for analytical convenience. Note that the dependence of the right hand side of Equation (11) on $\vec{x}_{0:t-1}^{j}$ does not contradict the factorization order of Equation (9), because it depends only on latent variables from timesteps prior to t. Furthermore, we choose the following proposal variational posterior:

$$q(\vec{g}_{:}^{:},\vec{x}_{:}^{:},\vec{v}_{:}^{:},,\vec{f}_{:}^{:,:},\vec{u}_{:}^{:,:}) = p(\vec{g}_{:}^{:}|\vec{x}_{:}^{:},\vec{v}_{:}^{:})q(\vec{v}_{:}^{:})p(\vec{f}_{:}^{:,:}|\vec{u}_{:}^{:,:})q(\vec{x}_{:}^{:})q(\vec{u}_{:}^{:}) \tag{12}$$

with $p(\vec{g}_{:}^{:}|\vec{x}_{:}^{:},\vec{v}_{:}^{:})$ given by Equation (7), $p(\vec{f}_{:}^{:,:}|\vec{u}_{:}^{:,:})$ by Equation (11) and $q(\vec{x}_{:}^{:})$ by Equation (3). The densities $q(\vec{v})$ and $q(\vec{u})$ are unconstrained except for normalization. With these distributions, we derive the standard free-energy ELBO [36], denoting $\Theta = (\vec{x}_{:}^{:},\vec{u}_{:}^{:,:},\vec{f}_{:}^{:,:},\vec{v}_{:}^{:},\vec{g}_{:}^{:})$:

$$\log p(\vec{y}_{:}^{:}) \geq \mathcal{L}(\Theta) = \int d\Theta q(\Theta) \log\left(\frac{p(\vec{y}_{:}^{:},\Theta)}{q(\Theta)}\right) \tag{13}$$

exploiting the assumption that the IPs $\vec{r}_{:}^{i}$ and IVs $\vec{v}_{:}^{i}$ are sufficient statistics for the function values $\vec{g}_{:}^{i}$. As we explain in detail in Appendix C, after canceling common factors in the variational posterior (Equation (12)) and the joint model density (Equation (5), cf. [16]), we find that the ELBO can be decomposed into one summand per part that describes the quality of the kinematics mapping (latent-to-observed) $\mathcal{L}_{kin}^{i}$, and one summand for the dynamics $\mathcal{L}_{dyn}$:

$$\mathcal{L}(\Theta) = \sum_{i=1}^{M} \mathcal{L}_{kin}^{i} + \mathcal{L}_{dyn} \tag{14}$$

where

$$\mathcal{L}_{kin}^{i} = \sum_{d=1}^{D} \int d\vec{x}_{:}^{i}\, d(\vec{v}_{:}^{i})_d\, d(\vec{g}_{:}^{i})_d\; p((\vec{g}_{:}^{i})_d|\vec{x}_{:}^{i},(\vec{v}_{:}^{i})_d) q(\vec{x}_{:}^{i}) q((\vec{v}_{:}^{i})_d) \log \frac{p((\vec{y}_{:}^{i})_d|(\vec{g}_{:}^{i})_d)}{q((\vec{v}_{:}^{i})_d)}. \tag{15}$$

is—up to the Shannon entropy of approximating posterior of the latent dynamics variables $H(q(\vec{x}_{:}^{i})) = -\int d\vec{x}_{:}^{i} q(\vec{x}_{:}^{i}) \log\left(q(\vec{x}_{:}^{i})\right)$—equal to the Bayesian GPLVM ELBO of [16]. The remaining integral

$$\begin{aligned}\mathcal{L}_{dyn} = &\int d\vec{u}_{:}^{:,:}\; q(\vec{u}_{:}^{:,:}) \left[\sum_{t=2}^{T} \int d\vec{x}_{1:t}\; q(\vec{x}_{1:t}) \left(\int d\vec{f}_t^{:,:}\; p(\vec{f}_t^{:,:}|\vec{f}_{2:t-1}^{:,:},\vec{x}_{0:t-1},\vec{u}_{:}^{:,:}) \log p(\vec{x}_t|\vec{f}_t^{:,:},\alpha^{:,:})\right)\right] \\ &+ \int d\vec{u}_{:}^{:,:} q(\vec{u}_{:}^{:,:}) \log \frac{p(\vec{u}_{:}^{:,:})}{q(\vec{u}_{:}^{:,:})} + \int d\vec{x}_{0:1} q(\vec{x}_{0:1}) \log p(\vec{x}_{0:1}) + H(q(\vec{x}_{:}^{:}))\end{aligned} \tag{16}$$

is derived in detail in Appendix C. Briefly, we use the assumption that the IPs and IVs $\vec{z}_{:}^{j,i}$ and $\vec{u}_{:}^{j,i}$ are sufficient statistics for the function values $\vec{f}_t^{j,i}$. Optimizing with respect to $q(\vec{u}^{:,:}:)$ can be carried out in closed form using variational calculus and yields

$$\mathcal{L}_{dyn}(\Theta) \geq \log \int p(\vec{u}_{:}^{:}) \exp(\mathcal{C}(\vec{u}_{:}^{:})) d\vec{u}_{:}^{:} + H(q(\vec{x}_{:}^{:})) \tag{17}$$

where $\mathcal{C}(\vec{u}_{:}^{:})$ is given by Equation (A31). The inequality is due to the sufficient statistics assumption, which introduces another approximation step that lower-bounds $\mathcal{L}_{dyn}$. We now have all the ingredients to compute the ELBO for the whole model, and learn it.

5. Results

We used the machine-learning framework `Theano` [37] for automatic differentiation in `Python 2.7` (Python Software Foundation. Python Language Reference, version 2.7. Available at http://www.python.org) to implement the model, and learned via optimization of the ELBO with the `scipy.optimize.fmin_l_bfgs_b` routine [38]. Latent space trajectories were initialized with PCA. We obtained the best ELBOs by first optimizing all parameters jointly, followed by a blocked optimization procedure. We optimize three groups of parameters: latent points and variances, kernel parameters and couplings, and IPs. The number of iterations of the blocked procedure depended on the application; we provide details in the sections below.

The advantage of the sparse approximations in the vCGPDM is that memory consumption of the model is greatly reduced. However, this approximation might also introduce errors, along with the fully factorized latent posterior (Equation (3)). We tried to quantify these errors in a cross-validatory model comparison, and in a human perception experiment.

5.1. Synthetic Data

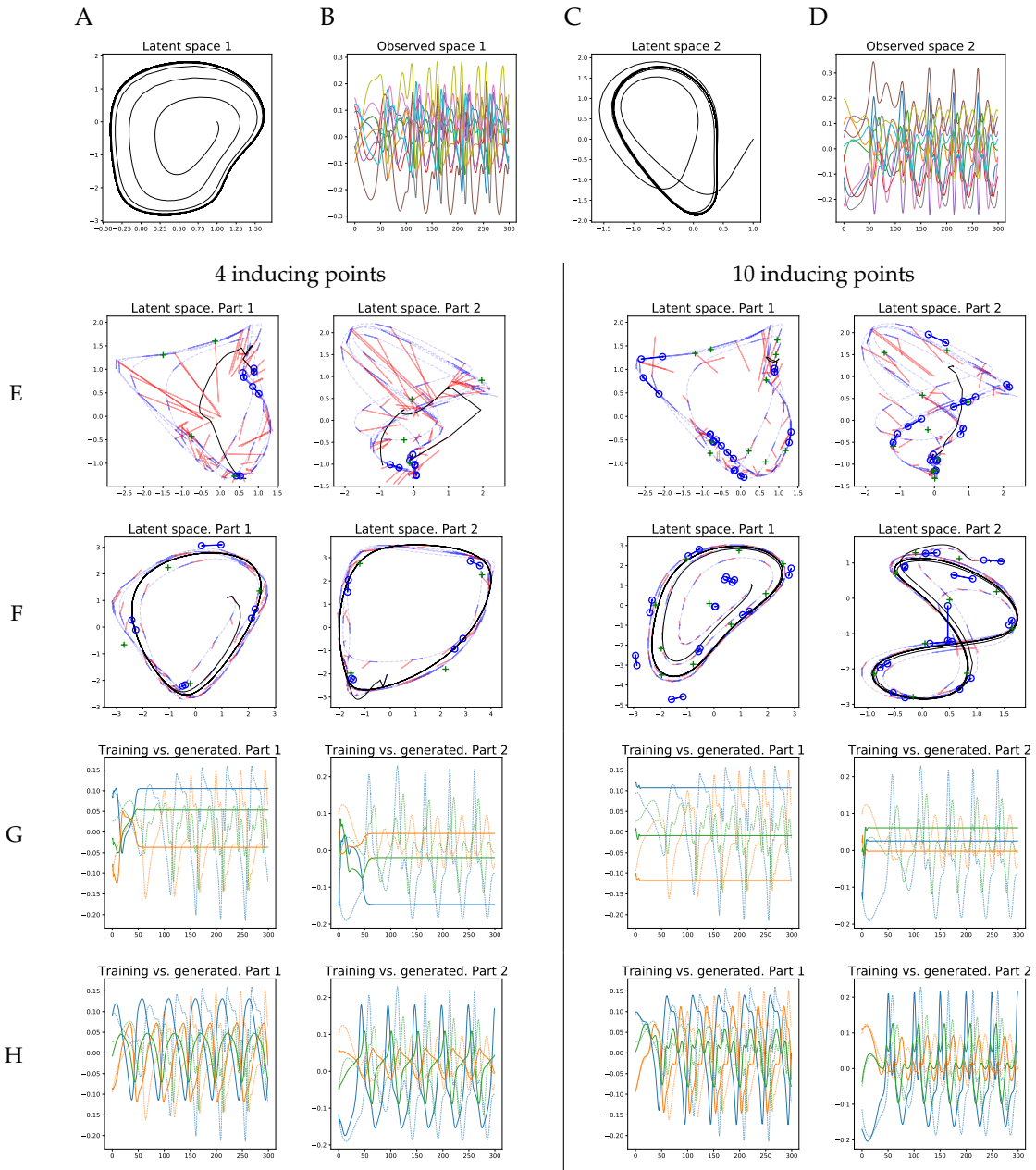
We demonstrate the learning of coupled dynamical systems on a synthetic dataset. First, we draw two dynamics transition functions $g^1, g^2 \in \mathbb{R}^2 \to \mathbb{R}^2$ from a $\mathcal{GP}$ with an RBF kernel, and then we generate latent trajectories according to:

$$\vec{x}_t^1 = g^1(\vec{x}_{t-1}^1) \tag{18}$$

$$\vec{x}_t^2 = 0.1 g^1(\vec{x}_{t-1}^1) + 0.9 g^2(\vec{x}_{t-1}^2) \tag{19}$$

at $T = 300$ timepoints. Thus, we get two first-order, coupled latent dynamical systems, each of dimensionality 2. The trajectory in Latent Space 1 is independent of Latent Space 2, whereas Latent Space 2 is weakly coupled to Latent Space 1. Then, for each of the two parts, we draw 10 observed trajectories from another two RBF GPs with inputs on the latent trajectories. The latent trajectories are shown in Figure 3A,C. Figure 3B,D displays the corresponding observed trajectories. We learned a second-order vCGPDM from these data, iterating the blocked optimization until convergence of the ELBO to machine precision. We chose a second order system for this learning example, because the human movement models in the following are second-order vCGPDMs, too.

The results are shown in Figure 3E–H. Plots on the left half were generated with four IPs, plots in the right half with ten IPs. Figure 3E displays the initial positions of the dynamics IPs (blue circles) at the beginning of learning, connected circles form one second-order IP. Green crosses are the kinematics IPs (latent-to-observed mapping). Initial latent points (dashed blue lines) were obtained from the first two PCA components of the training data. Blue and red line segments are examples of the dynamics mapping: the end-points of the blue segments are the inputs, the distal endpoint of the red segment is the output. As one might expect, the initial conditions do not describe a dynamics which can produce trajectories resembling those in the training data: the black line is the mean latent trajectory, and Figure 3G shows the corresponding observable trajectories.



$\alpha^{i,j}$	$j=1$	$j=2$
$i=1$	$\alpha^{1,1}=0.092$	$\alpha^{1,2}=0.55$
$i=2$	$\alpha^{2,1}=4.9$	$\alpha^{2,2}=0.27$
	initial	final
ELBO:	$-9.0\cdot10^{3}$	$1.0\cdot10^{4}$
MSE:	$4.7\cdot10^{-3}$	$1.3\cdot10^{-3}$

$\alpha^{i,j}$	$j=1$	$j=2$
$i=1$	$\alpha^{1,1}=9.6\cdot10^{-3}$	$\alpha^{1,2}=0.023$
$i=2$	$\alpha^{2,1}=10$	$\alpha^{2,2}=0.011$
	initial	final
ELBO:	$-8.1\cdot10^{3}$	$1.3\cdot10^{4}$
MSE:	$5.7\cdot10^{-3}$	$6.1\cdot10^{-4}$

Figure 3. Synthetic training data. (**A**–**D**) Two-dimensional latent dynamics trajectories and corresponding observed 10-dimensional time series. Part 2 is weakly influenced by Part 1, Part 1 is not influenced by Part 2 (see Equation (18)). (**E**) Initial positions of second-order dynamics IPs (connected blue circles) and latent-to-observed IPs (green crosses). Line segments are examples of dynamics mapping inputs (endpoints of blue segments) and values (distal endpoints of red line segments). Black line: mean trajectory, generated by iterating the mean dynamics mapping from the same starting point as in (**F**) Latent space after learning. (**G**) Generated observable time series before learning (solid) and training data (dashed). (**H**) Generated time series (solid) after learning. Only three of the ten observable trajectories are presented for clarity. (**Bottom**) Learned couplings (see Equation (1)); ELBOs and MSEs rounded to two significant digits. Couplings $\alpha^{i,j}$ reflect the dependency structure between parts: Part 1 is not driven by Part 2, but influences Part 2.

After learning, the latent trajectories appear to have a limit cycle (black line, Figure 3F), which is required to reproduce the training data. Furthermore, note that the inducing points align with that cycle, and the example mappings (blue and red line segments) indicate clearly how the latent trajectory is formed by iterating the GP mapping. Cross coupling GP mappings omitted for clarity. Note that the vCGPDM with ten IPs can model more complex latent dynamics manifolds than the four-IP vCGPDM. The observable trajectories (Figure 3H) look very similar to the training data up to a small phase shift, particularly for the 10 IP model. This observation is confirmed by the reduced mean squared trajectory error (MSE) between generated and training data after learning, which was evaluated after dynamic time warping [39] of the generated trajectories onto the training data. The MSEs are listed in the table at the bottom of Figure 3, where "final" indicates the values after learning, while "initial" indicates the values at the onset of learning after the latent space trajectories had been initialized to the first two PCA components of the training data. That learning was successful is also indicated by the increased final ELBO, which is higher for the 10 IP model.

We also provided the learned coupling αs in this table. Recall that a low (high) α means a large (small) influence of the corresponding part on the dynamics. The dependency structure between the latent spaces was correctly identified during learning: $a^{2,1} \gg \alpha^{1,1}$, i.e., Part 2 has almost no influence on Part 1. In contrast, $\alpha^{1,2} \approx 2\alpha^{2,2}$, which indicates that Part 1 weakly controls Part 2.

5.2. Human Movement Data

Model comparisons and psychophysical tests were carried out on human movement data. We employed a 10-camera PhaseSpace Impulse motion capture system, mapped the resulting position data onto a skeleton with 19 joints and computed joint angles in exponential-map representation, yielding a total of 60 degrees of freedom. Five walking-only and four walking + waving sequences each were used to train the models, as well as ten movements where the human participants were seated and passed a bottle from one hand to the other. Dynamical models were initialized with starting conditions taken from the training data. The blocked optimization was run for at most four iterations, which was enough to ensure convergence. It was terminated earlier if ELBO values did not change within machine precision between two subsequent iterations. Furthermore, we recorded another nine walking sequences for catch trials during the perception experiment, to rule out memorization effects. Generated and recorded sequences were rendered on a neutral avatar. Examples of stimuli, for different numbers of IPs, can be found in the movie `example_stimuli.mov` in the Supplementary Materials.

5.3. Variational Approximations are Better than MAP

We performed cross-validatory model comparison on the following datasets: walking, walking + waving and passing-a-bottle. Examples of these data are shown in the movies in the Supplementary Materials: `S1_example_stimuli.mov` and `S4_pass_the_bottle.mkv`. We performed four-, five- and ten-fold crossvalidation, the number of folds was dictated by the dataset size. We were trying determine how the sparsely parameterized vCGPDM performs in comparison to the full CGPDM, and several other MP models from the literature. Held-out data were always one complete trial. Models were trained on the remaining data and the generated trajectory was compared to the held-out one. Cross-validation score was the mean-squared error (MSE) of the kinematics after dynamic time warping [39] of trajectories generated by initializing the model to the first two frames of a held-out trial onto the complete held-out trial. We used dynamic time warping to compensate a slight phase difference in generated motions, which would otherwise lead to a much larger and uninformative MSE. We compared the following models:

- a GPDM with maximum-a-posteriori (MAP) estimation of the latent variables [25], called MAP GPDM in Figure 4;
- a fully marginalized two-part (upper/lower body) CGPDM with MAP estimation of the latent variables [14], called MAP CGPDM U+L;

- a three-part CGPDM model (left hand, right hand, and body) for the non-periodic "passing a bottle" dataset;
- their variational counterparts, vCGPDM 3-part, vCGPDM U+L and vGPDM;
- temporal MPs (TMP, instantaneous linear mixtures of functions of time) [9]; and
- DMPs [12].

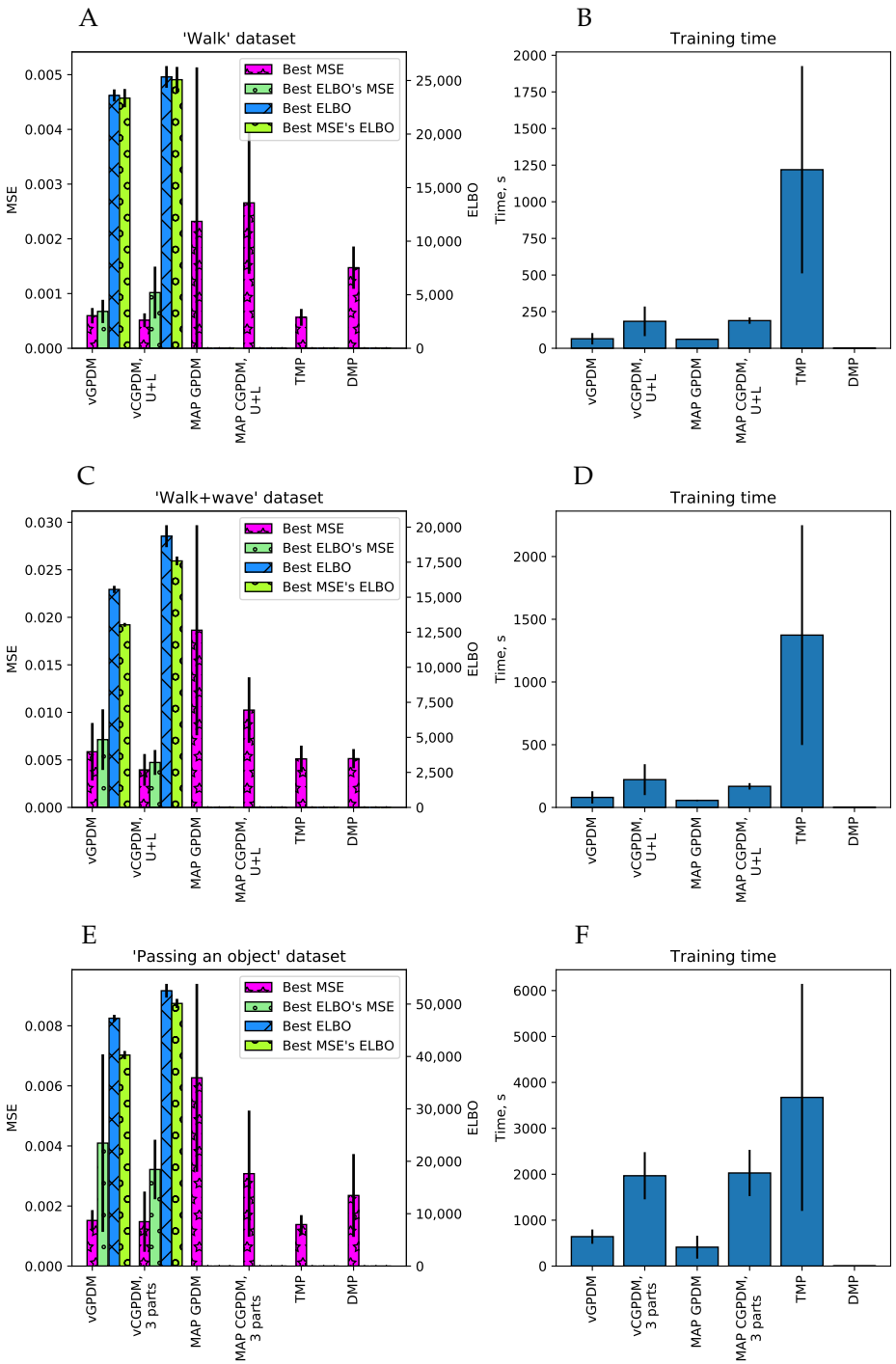


Figure 4. Model comparison results. We plotted the average squared kinematics error on held-out data after dynamic time warping (MSE) and the variational lower bound on the model evidence (ELBO, Equation (A22)), where available, accompanied with corresponding model training time. Error bars are standard errors of the mean. (**A**,**B**) Walking dataset; (**C**,**D**) walking + waving dataset; and (**E**,**F**) "passing a bottle" dataset. Low MSE and high ELBO are better. For details, see text. Figure partially adapted from [19].

All latent spaces were three-dimensional. We tried 4–30 latent-to-observed IPs and 2–30 dynamics IPs. The MSE optima were near 10–15 IPs for both the walking and the walking + waving datasets, and near eight IPs for the "passing a bottle". MAP GPDM and MAP CGPDM learning do not use any approximations or inducing points; they are the full GPs with covariance matrices $\mathbf{K} \in \mathbb{R}^{T*T}$.

For the TMPs, we used up to 10 primitives; the MSE optimum was located at approximately six. For the DMPs, we used between 1 and 50 basis functions, and the lowest MSE was found around 15.

The results are plotted in Figure 4. Generally, the walking + waving movement is more difficult to reproduce for all models than walking only: the MSE of the latter is lower than that of the former, and the ELBO is higher. This indicates that the latter is a more complex movement, see also the movie `modular_primitives.avi` in the online Supplementary Materials. The two-part vCGPDM reaches the lowest MSE compared to all other models. Clearly, it is better than the full-capacity (no IPs) MAP models, which means that the extra effort of developing of a variational approximation which explicitly represents an approximation to the latent states' posterior and needs to store only $\approx$10 IPs rather than $\approx 10^4$ data points was well spent. In addition, the best ELBO's MSE (that is, the MSE at the maximum of the ELBO) is a fairly good predictor of the best MSE, which justifies our simple variational approximation for model selection.

The vCGPDM U+L outperforms the vGPDM particularly on the "walking + waving" dataset. This shows the usefulness of having modular, coupled dynamics models when the (inter)acting (body)parts execute partially independent movements.

The vCGPDM with three parts for "passing a bottle" does not show a clear advantage over the monolithic model in the cross-validation test, and is on par with the TMP model. However, dynamics factorization did not affect the performance either. This may be indicative for the strong coupling necessary to successfully pass an object from one hand to the other. Such a strong coupling is parsimoniously expressed by having a single latent dynamical system drive all observable degrees of freedom.

The timing results show that the training times of the vCGPDM are usually less than 15 min. Error bars are standard deviations, estimated across all numbers of IPs and cross-validation splits. The rather large training time for the TMP model is due to the implementation from [9] which optimizes a rather large covariance matrix between all MPs.

5.4. A Small Number of IPs Yields Perceptually Convincing Movements

We conducted a psychophysical experiment to quantify the perceptual validity of the generated movements. More specifically, we investigated the model complexity required for perceptually convincing movements.

Experiment: Thirty-one human observers (10 male, mean age: 23.8 ± 3.5a) participated in a two-alternative forced-choice task to distinguish between natural and generated movements (see Figure 5 for an example of an experimental trial). Natural movements consisted of 15 walking movements. The artificial movements were generated by a two-part (upper/lower body) vCGPDM. We used 2–16 dynamics IPs and 4–16 latent-to-observed IPs. We chose these numbers based on pilot tests to span the range from clearly unnatural to very natural looking movements. To test whether participants simply memorized the 15 natural stimuli during the experiment, we added 10 catch trials in the last quarter of the experiment where previously unused natural movements were tested against the known natural stimuli. The trial sequence was randomized for every subject. All experimental procedures were approved by the local ethics commission.

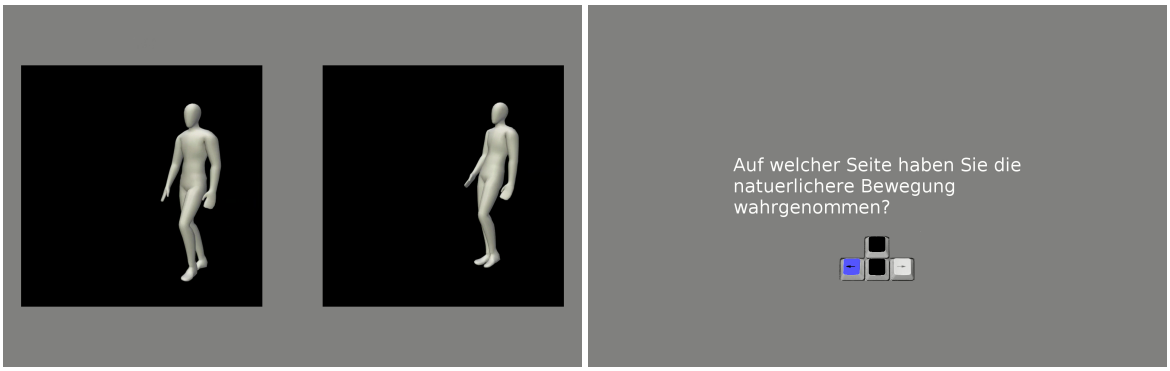


Figure 5. Psychophysical Experiment. In each trial, a natural and a generated movement were simultaneously presented to participants (**left**). After presentation, they used the arrow keys to choose the movement perceived as more natural (**right**). There was no time limit on the response, but typically participants responded quickly (less than 1 s). After the response, people were asked to fixate a cross in the middle of the screen, which appeared for a second. The length of the stimuli was 1.8 s, with a total of 1170 presentations. A video of the experiment called `S2_experiment_demo.avi` is provided in the Supplementary Materials.

Results: We computed the confusion rate, i.e., the frequency of choosing the model-generated movement as more natural across all participants as a function of the number of IPs for the dynamics and latent-to-observed mappings. Optimally, we might expect this rate to approach $\frac{1}{2}$ when the generated movements are indistinguishable from the natural ones. We investigated if the confusion rate approached this limit, how it depends on the mean-squared residual error on the training data, and how this error is connected to the ELBO. The results are plotted in Figure 6. We also fitted the confusion rate data with a logistic sigmoid $\frac{0.5}{1+\exp(a \cdot MSE + c)}$ (solid line in Figure 6A), and the MSE with an exponential function (solid line in Figure 6, right). Each data point represents one combination of dynamics/latent-to-observed IP numbers, indicated by width and height of the ellipses. Clearly, confusion rate increases fairly monotonically with decreasing MSE, as indicated by the good logistic sigmoid fit. Furthermore, models with more IPs also tend to yield higher confusion rates. A sufficient number (>10) dynamics IPs is more important than a large number of latent-to-observed IPs, which can be seen by the very narrow ellipses in the region with high MSE, and many wider ellipses in the lower MSE part of the figure. A similar observation can be made about the relationship between ELBO and MSE (Figure 6B). It indicates that ELBO is already a good predictor for the model performance. For a very small number of dynamics IPs, increasing the number of latent-to-observed IPs does not decrease the MSE as much as increasing the dynamics IPs does. Moreover, note that the relationship between MSE and ELBO becomes fairly monotonic when ELBO > 28,500, which is where human perceptual performance can be predicted from ELBO. While the confusion rate has not quite reached its theoretical maximum in our experiment, these results are evidence that human perceptual expectations can be nearly met with very compactly parameterized MP models (Figure 6C,D). Moreover, good dynamics models seem to outweigh precise kinematics. We found no evidence for stimulus memorization from the confusion rates of the catch trials.

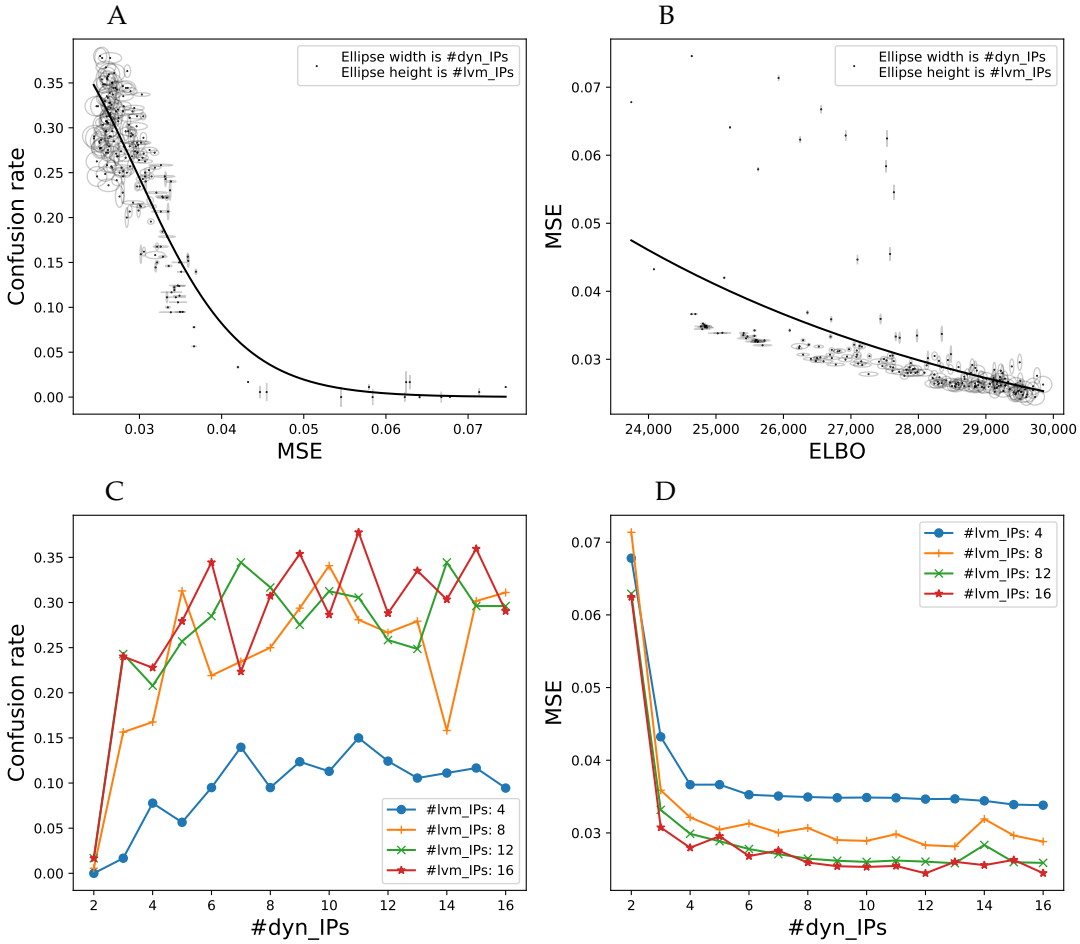
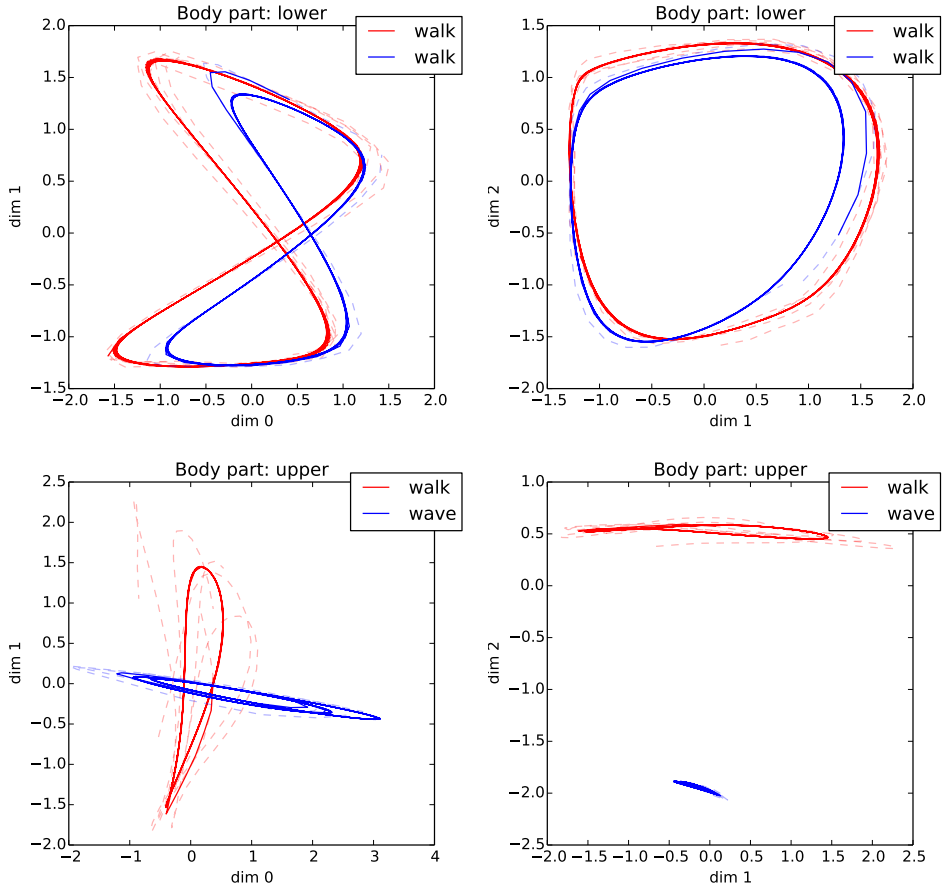


Figure 6. (**A**) Confusion rate between natural and vCGPDM-generated stimuli as a function of mean-squared residual error (MSE) on the training data, averaged across all participants. Each data point represents one combination between number of IPs/IVs for the latent-to-observed mapping (indicated by ellipse height) and number of IPs/IVs for the dynamics mappings (ellipse widths). A confusion rate of 0.5 indicates that human observers are not able to distinguish replays of real movements from model-generated counterparts. The vCGPDM is approaching this limit from below for a fairly small number of IPs/IVs. Solid line: fit with logistic sigmoid function. (**B**) Relationship between training MSE and ELBO. Solid line: fit with exponential function. Additional dynamics IPs contribute more to the reduction of the MSE than latent-to-observed IPs. MSE and therefore confusion rate can be predicted well from ELBO if ELBO > 28,500. (**C**,**D**) Influence of number of dynamics IPs on the confusion rate and MSE, respectively, for a selected number of latent-to-observed IPs. The confusion rate has a broad maximum around 8–12 dynamics IPs, whereas the MSE has a shallow minimum at that location.

5.5. Modularity Test

Next, we examined if the intended modularization of our model can be used to compose novel movements from previously learned parts. We trained a vCGPDM consisting of one part for the lower body (below and including pelvis), and a second part for the upper body. Twenty-five IPs for the latent-to-observed mapping of each part were shared across all movements. The walking MP, parameterized by 16 IPs for the lower-body dynamics and the lower-to-upper mappings, was also shared. We used a different set of 16 IPs for the upper body MPs between arm-swing and waving. Furthermore, the coupling $\alpha^{j,i}$ were learned anew for each combination of upper/lower MPs. The resulting latent space trajectories are plotted in Figure 7. All generated trajectories (solid lines) are on average close to the training data (dashed lines). While the walking trajectories

for the lower body are very similar for the two movements, the upper body trajectories clearly differ. Movements generated from this model are very natural (see video `S3_modular_primitives.mov` in the Supplementary Materials). This is a first demonstration that the vCGPDM with non-marginalized couplings can be used to learn a library of compactly parameterized MPs, from which novel movements can be produced with little additional memory requirements (i.e., new coupling $\alpha^{j,i}$ only).



Dataset		vCGPDM	Modular vCGPDM
Walking	Cross-validation MSE	0.0042($\pm$0.0035)	0.0042($\pm$0.0017)
	Training time, s	1948($\pm$225)	6289($\pm$709)*
Walking and waving	Cross-validation MSE	0.0050($\pm$0.0006)	0.0057($\pm$0.0013)
	Training time, s	1501($\pm$236)	6289($\pm$709)*

Figure 7. (**Top**) Modularity example. Shown are 2D projections of generated 3D latent space trajectories (solid) and training data (dashed). Blue: walk + wave movements; red: walk + normal arm swing. Dynamics IPs re-used across movements for lower body. (**Bottom**) Cross-validation MSEs of non-modular and modular vCGPDM. Modular vCGPDM was trained on the combined dataset; training time (*) is shared between both movement datasets.

For a quantitative evaluation, we looked at the leave-one-out cross-validation MSEs of the vCGPDM trained on datasets separately and modular vCGPDM trained on both datasets (see Figure 7, bottom). Within the standard errors, MSEs are equal, indicating that modular re-use of previously trained components does not necessarily sacrifice accuracy, while reducing storage requirements. Training time for the modular vCGPDM is larger due to the learning of the combined dataset and optimizing the couplings afterwards. This time would be amortized if more compositional movements were learned, where previously learned parts could be reused.

6. Conclusions

The vCGPDM, a full variational approximation to the CGPDM, allows for learning a deterministic approximation of latent space trajectories, and compactly parameterizing dynamics and kinematics mappings. First, we showed that the sparsely parameterized vCGPDM outperforms the full-capacity, monolithic CGPDM employing MAP to approximate the latent dynamics posterior. It also surpasses other current MP models; we speculate that this is accomplished by its learnable dynamics.

Second, we demonstrated that our compact representation of the latent space dynamics, and of the latent-to-observed mapping, enables the model to generate perceptually convincing full-body movements with a fairly small number of IPs To our knowledge, a systematic investigation of the number of IPs needed for perceptual plausibility had not been done before, albeit more monolithic models were in the focus of earlier studies [27,34,40]. Moreover, we demonstrated that a high enough ELBO can be used to predict average human classification performance, which might allow for an automatic model selection process when training the model on large databases. Within the range of IPs which we tested, the ELBO was still increasing with their number. We chose that range because we wanted to see how few IPs would still lead to perceptually indistinguishable movements. Due to experimental time constraints, we did not investigate perceptual performance at the point where the ELBO begins to decrease with increasing IPs (i.e., the approximately optimal model), but we plan to do that in the future.

Third, we showed that the model can be employed in a modular fashion, using one lower-body dynamics model, and coupling it to two different models for the upper body. Note that the lower-to-upper coupling function was the same for the two upper-body models. Each of these models, including the coupling functions to the other model parts, may therefore be viewed as a modular MP that is parameterized compactly by a small number of IPs and values. This sparse parameterization allows us to infer modular MPs from a large collection of movements, and investigate their composition. To generate complex movement sequences, we will put a switching prior on top of the dynamical models, as in [29].

We are currently researching sensorimotor primitives, i.e., MPs that can be used to predict sensory input and be controlled by it via conditioning. This conditioning can take place on at least two timescales: a short one (while the MP is running), thus effectively turning the MPs into flexible control policies, such the probabilistic MPs described by Paraschos et al. [41], and a long timescale, i.e., the planning of the movement. This could be implemented by learning a mapping from goals and affordances onto the coupling weights, comparable to the DMPs with associative skill memories [42]. There is evidence that humans modulate the coupling between their MPs during the planning stage: whole-body posture changes have been observed in anticipation of reaching for a goal object in a known location, even if the object is currently invisible [43].

Lastly, we note that our CGPDM could be used as a flexible policy model for PILCO-style reinforcement learning (Probabilistic Inference for Learning Control, [44]). PILCO requires a dynamics model that can propagate uncertainties through time; the vCGPDM is able to do that. Thus, our model could be used as a lower dimensional dynamics model which can capture the dependencies between observable variables via latent space uncertainties.

Supplementary Materials: The following are available online http://www.mdpi.com/1099-4300/20/10/724/s1, Video S1: example_stimuli, Video S2: experiment_demo, Video S3: modular_primitives, Video S4: pass_the_bottle.

Author Contributions: D.V. and D.E. conceived vCGPDM model. D.V. derived, implemented and tested models. B.K. and D.E. conceived psychophysical experiment. B.K. conducted experiment and evaluated data. All authors wrote paper.

Funding: This work was supported by the Deutsche Forschungsgemeinschaft under DFG-IRTG 1901 "The Brain in Action" and DFG-SFB-TRR 135 project C06.

Acknowledgments: We thank Olaf Haag for help with rendering the movies, and Björn Büdenbender for assistance with MoCap. Open access costs were paid by the University Library of Marburg and the DFG.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

MP	movement primitive
DMP	dynamical movement primitive
$\mathcal{GP}$	Gaussian process
GPDM	Gaussian process dynamical model
CGPDM	coupled Gaussian process dynamical model
vCGPDM	variational coupled Gaussian process dynamical model
IP	inducing point
IV	inducing value
MSE	mean squared error

Appendix A. Exact Variational Optimization of Parts of the ELBO

While optimizing the full variational posterior in augmented Gaussian processes models with respect to the IVs, the following type of term appears several times in the ELBO equation:

$$\begin{aligned}\mathcal{R}(q(\vec{u}), r(\vec{v})) &= \int q(\vec{u}) \left(f(r(\vec{v}), \vec{u}) + \log \tfrac{p(\vec{u})}{q(\vec{u})} \right) d\vec{u} \\ &= \int q(\vec{u}) f(r(\vec{v}), \vec{u}) d\vec{u} + \int q(\vec{u}) \log p(\vec{u}) d\vec{u} - \int q(\vec{u}) \log q(\vec{u}) d\vec{u}\end{aligned} \tag{A1}$$

To simplify the optimization of such terms, we would like to carry out the optimization with respect to the density $q(\vec{u})$ analytically to remove the dependency on $q(\vec{u})$. Note that we allow only $q(\vec{u}), r(\vec{v})$ to vary, while the functions $f(r(\vec{v}), \vec{u})$ and $p(\vec{u})$ are assumed to be fixed. To this end, we calculate for the optimal variational $q^*(\vec{u})$ in the above equation. This approach was suggested in [23], however, it is not well described there. Here, we give an extended derivation. A necessary condition for maximality is a vanishing functional derivative under the constraint that the density $q(\vec{u})$ is normalized to one:

$$\int q(\vec{u}) d\vec{u} - 1 = 0 \tag{A2}$$

which is fulfilled at the stationary points of the Lagrangian

$$\mathcal{X}(q(\vec{u}), r(\vec{v})) = \mathcal{R}(q(\vec{u}), r(\vec{v})) + \lambda \left(\int q(\vec{u}) d\vec{u} - 1 \right) \tag{A3}$$

where λ is chosen so that Equation (A2) holds. Taking the functional derivative of $\mathcal{X}(q(\vec{u}), q(\vec{v}))$ and setting it to zero yields

$$\frac{\delta \mathcal{X}(q(\vec{u}), q(\vec{v}))}{\delta q(\vec{u})} = f(r(\vec{v}), \vec{u}) + \log p(\vec{u}) - \log q(\vec{u}) - 1 + \lambda = 0 \tag{A4}$$

and therefore, denoting $Z = \exp(-\lambda + 1)$

$$q^*(\vec{u}) = \exp(f(r(\vec{v}), \vec{u}) + \log p(\vec{u}) - 1 + \lambda) \tag{A5}$$

$$q^*(\vec{u}) = \frac{1}{Z} p(\vec{u}) \exp(f(r(\vec{v}), \vec{u})) \tag{A6}$$

$$Z = \exp(-\lambda + 1) = \int p(\vec{u}) \exp(f(r(\vec{v}), \vec{u})) d\vec{u} \tag{A7}$$

Substituting the optimal $q^*(\vec{u})$ into the original term, we get:

$$\begin{aligned}\mathcal{R}(r(\vec{v})) &= \int \tfrac{1}{Z} p(\vec{u}) \exp(f(r(\vec{v}),\vec{u})) \left(f(r(\vec{v}),\vec{u}) + \log \frac{p(\vec{u})}{\frac{1}{Z} p(\vec{u}) \exp(f(r(\vec{v}),\vec{u}))} \right) d\vec{u} \\ &= \int \tfrac{1}{Z} p(\vec{u}) \exp(f(r(\vec{v}),\vec{u})) \left(\log \frac{p(\vec{u}) \exp(f(r(\vec{v}),\vec{u}))}{\frac{1}{Z} p(\vec{u}) \exp(f(r(\vec{v}),\vec{u}))} \right) d\vec{u} \\ &= \log(Z) \tfrac{1}{Z} \int p(\vec{u}) \exp(f(r(\vec{v}),\vec{u})) d\vec{u} \\ &= \log \int p(\vec{u}) \exp(f(r(\vec{v}),\vec{u})) d\vec{u}\end{aligned} \tag{A8}$$

This is the optimized version of Equation (A1), which depends only on $r(\vec{v})$.

Appendix B. ARD RBF Kernel Ψ Statistics. Full Covariance Variational Parameters Case.

During the computation of the ELBO, it is necessary to evaluate expected values of Gaussian process kernel functions under the variational posterior distributions. Here, we derive these expectations, referred to as Ψ statistics in the literature [16], for the type of kernel we used in this paper: an automatic relevance determination, squared exponential kernel (ARD RBF). The ARD RBF kernel is defined as:

$$k(\vec{x},\vec{x}') = \sigma_f^2 \exp\left(-\frac{1}{2} \sum_{q=1}^{Q} \frac{((\vec{x})_q - (\vec{x})'_q)^2}{\lambda_q} \right) \tag{A9}$$

where λ_q are the ARD factors, σ_f^2 is the variance of the kernel and Q is the dimensionality of $\vec{x}$. In matrix notation:

$$\lambda = \mathrm{diag}(\lambda_1 \dots \lambda_Q) \tag{A10}$$

$$k(\vec{x},\vec{x}') = \sigma_f^2 \exp\left(-\frac{1}{2} (\vec{x}-\vec{x}')^T \lambda^{-1} (\vec{x}-\vec{x}') \right) \tag{A11}$$

Let $\vec{x}$ be a random variable drawn from multivariate Gaussian distributions with mean $\vec{\mu}$ and covariance matrix S. Consider the following form of the approximate variational posterior distribution of $\vec{x}$:

$$q(\vec{x}) = \mathcal{N}(\vec{x}_n | \vec{\mu}, S) \tag{A12}$$

The Ψ_0 statistic is the expectation of the kernel for two identical arguments, which is easy to calculate:

$$\begin{aligned}\Psi_0 &= \int k(\vec{x}_n,\vec{x}_n) \mathcal{N}(\vec{x}_n|\vec{\mu}_n,S_n) d\vec{x}_n \\ &= \int \sigma_f^2 \mathcal{N}(\vec{x}_n|\vec{\mu}_n,S_n) d\vec{x}_n \\ &= \sigma_f^2 \int \mathcal{N}(\vec{x}_n|\vec{\mu}_n,S_n) d\vec{x}_n \\ &= \sigma_f^2\end{aligned} \tag{A13}$$

The Ψ_1 statistic is the expectation with respect to one kernel argument, given that the other is constant:

$$\begin{aligned}\Psi_1 &= \int k(\vec{x},\vec{z}) \mathcal{N}(\vec{x}|\vec{\mu},S) d\vec{x} \\ &= \int \sigma_f^2 \exp\left(-\tfrac{1}{2} (\vec{x}-\vec{z})^T \lambda^{-1} (\vec{x}-\vec{z}) \right) \mathcal{N}(\vec{x}|\vec{\mu},S) d\vec{x}\end{aligned} \tag{A14}$$

To evaluate the integral, complete the ARD RBF kernel to a scaled Gaussian distribution with covariance matrix λ and mean $\vec{z}$:

$$\begin{aligned}\Psi_1 &= \sigma_f^2 \int \tfrac{Z(\lambda)}{Z(\lambda)} \exp\left(-\tfrac{1}{2} (\vec{x}_r-\vec{z})^T \lambda^{-1} (\vec{x}_r-\vec{z}) \right) \mathcal{N}(\vec{x}_n|\vec{\mu}_n,S_n) d\vec{x} \\ &= \sigma_f^2 \, Z(\lambda) \int \mathcal{N}(\vec{x}|\vec{z},\lambda) \mathcal{N}(\vec{x}|\vec{\mu},S) d\vec{x}\end{aligned} \tag{A15}$$

with $Z(\lambda) = (2\pi)^{Q/2}\sqrt{|\lambda|}$. The integral over the product of two Gaussians can be carried out to yield (see [45], Identity 371):

$$\begin{aligned}\Psi_1 &= \sigma_f^2\, Z(\lambda)\mathcal{N}(\vec{z}|\vec{\mu}, \lambda + S)\\ &= \sigma_f^2\, (2\pi)^{Q/2}\sqrt{|\lambda|}\frac{1}{(2\pi)^{Q/2}\sqrt{|\lambda + S|}}\exp\left(-\tfrac{1}{2}(\vec{z}-\vec{\mu})^T(\lambda+S)^{-1}(\vec{z}-\vec{\mu})\right)\\ &= \sigma_f^2\, \frac{\sqrt{|\lambda|}}{\sqrt{|\lambda + S|}}\exp\left(-\tfrac{1}{2}(\vec{z}-\vec{\mu})^T(\lambda+S)^{-1}(\vec{z}-\vec{\mu})\right)\\ &= \sigma_f^2\, \sqrt{\frac{\prod_{q=1}^{Q}\lambda_q}{|\lambda + S|}}\exp\left(-\tfrac{1}{2}(\vec{z}-\vec{\mu})^T(\lambda+S)^{-1}(\vec{z}-\vec{\mu})\right)\end{aligned} \tag{A16}$$

The Ψ_2 statistic integral, which correlates two kernel function values at different points $\vec{z}$ and $\vec{z}'$, can be solved in a similar manner: first, by collecting terms in the exponents and completing quadratic forms, and second, by the application of Identity 371 from [45]:

$$\begin{aligned}\Psi_2 &= \int k(\vec{x},\vec{z})k(\vec{z}',\vec{x})\mathcal{N}(\vec{x}|\vec{\mu}, S)d\vec{x}\\ &= (\sigma_f^2\, Z(\lambda))^2\ \int \mathcal{N}(\vec{x}|\vec{z},\lambda)\mathcal{N}(\vec{x}|\vec{z}',\lambda)\mathcal{N}(\vec{x}|\vec{\mu}, S)d\vec{x}\\ &= (\sigma_f^2 Z(\lambda))^2\ \int \mathcal{N}(\vec{z}'|\vec{z}, 2\lambda)\mathcal{N}(\vec{x}|\frac{1}{2}(\vec{z}'+\vec{z}), \frac{1}{2}\lambda)\mathcal{N}(\vec{x}|\vec{\mu}, S)d\vec{x}\\ &= (\sigma_f^2 Z(\lambda))^2\ \mathcal{N}(\vec{z}'|\vec{z}, 2\lambda)\int \mathcal{N}(\vec{x}|\frac{1}{2}(\vec{z}+\vec{z}'), \frac{\lambda}{2})\mathcal{N}(\vec{x}|\vec{\mu}, S)d\vec{x}\\ &= (\sigma_f^2 Z(\lambda))^2\ \mathcal{N}(\vec{z}'|\vec{z}, 2\lambda)\mathcal{N}(\vec{\mu}|\frac{1}{2}(\vec{z}+\vec{z}'), \frac{\lambda}{2} + S)\\ &= \sigma_f^4(2\pi)^Q(\prod_{q=1}^{Q}\lambda_q)\ \mathcal{N}(\vec{z}'|\vec{z}, 2\lambda)\ \mathcal{N}(\vec{\mu}|\frac{\vec{z}+\vec{z}'}{2}, \frac{\lambda}{2} + S)\end{aligned} \tag{A17}$$

For the case of a diagonal covariance matrix S the Ψ_2 statistic can be simplified further [16].

Appendix C. vCGPDM Dynamics ELBO Derivation

We now present a detailed derivation of the ELBO with a focus on the dynamics component. Assume we deal with M parts. We have $M \times M$ latent dynamics mappings, which are combined into M mappings with product of experts—multiplying and renormalizing the distributions from all parts' predictions about each part. Each of the $M \times M$ mappings $f^{j,i}()$ is augmented with IPs $\vec{z}_{:}^{\,j,i}$ and IVs $\vec{u}_{:}^{\,j,i}$, which are drawn out of the same $\mathcal{GP}$ priors as the mappings. For clarity, we omit spelling out the dependence of the IVs on the IPs in the following, and we ask the reader to remember that any distribution over IVs is implicitly conditioned onto the corresponding IPs. The full augmented joint distribution of the model, which is derived in Section 4 (Equation (5)), is:

$$\begin{aligned}p(\vec{y}_{:}, \vec{g}_{:}, \vec{x}_{:}, \vec{v}_{:}, \vec{f}_{:}^{\,:}, \vec{u}_{:}^{\,:}) &= p(\vec{y}_{:}|\vec{g}_{:})p(\vec{g}_{:}|\vec{x}_{:}, \vec{v}_{:})p(\vec{v}_{:})p(\vec{x}_{:}, \vec{f}_{:}^{\,:}|\vec{u}_{:}^{\,:})p(\vec{u}_{:}^{\,:})\\ &= \left[\prod_{i=1}^{M} p(\vec{y}_{:}^{\,i}|\vec{g}_{:}^{\,i})p(\vec{g}_{:}^{\,i}|\vec{x}_{:}^{\,i}, \vec{v}_{:}^{\,i})p(\vec{v}_{:}^{\,i})\right] p(\vec{x}_{:}, \vec{f}_{:}^{\,:}|\vec{u}_{:}^{\,:})p(\vec{u}_{:}^{\,:})\\ &= \left[\prod_{i=1}^{M}\left[\prod_{d=1}^{D_i} p((\vec{y}_{:}^{\,i})_d|(\vec{g}_{:}^{\,i})_d)p((\vec{g}_{:}^{\,i})_d|\vec{x}_{:}^{\,i}, (\vec{v}_{:}^{\,i})_d)p((\vec{v}_{:}^{\,i})_d)\right]\right]\\ &\times \left[\prod_{t=1}^{T}\left[\prod_{i=1}^{M} p(\vec{x}_t^{\,i}|\vec{f}_t^{\,:,i}, \alpha^{:,i})\right]\left[\prod_{i=1}^{M}\prod_{j=1}^{M} p(\vec{f}_t^{\,j,i}|\vec{f}_{1:t-1}^{\,j,i}, \vec{x}_{0:t-1}^{\,j}, \vec{u}_{:}^{\,j,i})\right]\right]\\ &\times \left[\prod_{i=1}^{M}\prod_{j=1}^{M} p(\vec{u}_{:}^{\,j,i})\right]\left[\prod_{i=1}^{M} p(\vec{x}_0^{\,i})\right]\end{aligned} \tag{A18}$$

The full proposal variational posterior is (Equation (12) in Section 4):

$$\begin{aligned} q(\vec{g}_{:}, \vec{x}_{:}, \vec{v}_{:}, \vec{f}_{:}^{:}, \vec{u}_{:}^{:}) &= p(\vec{g}_{:}|\vec{x}_{:}, \vec{v}_{:}) q(\vec{v}_{:}) p(\vec{f}_{:}^{:}|\vec{x}_{:}, \vec{u}_{:}^{:}) q(\vec{x}_{:}) q(\vec{u}_{:}^{:}) \\ &= p(\vec{g}_{:}|\vec{x}_{:}, \vec{v}_{:}) q(\vec{v}_{:}) \left[\prod_{t=1}^{T} \prod_{i=1}^{M} \prod_{j=1}^{M} p(\vec{f}_{t}^{j,i}|\vec{f}_{1:t-1}^{j,i}, \vec{x}_{0:t-1}^{j}, \vec{u}_{:}^{j,i}) \right] q(\vec{x}_{:}) q(\vec{u}_{:}^{:}) \end{aligned} \tag{A19}$$

Thus, the ELBO is given by:

$$\mathcal{L}(\Theta) = \int d\vec{g}_{:} \, d\vec{x}_{:} \, d\vec{v}_{:} \, d\vec{f}_{:}^{:} \, d\vec{u}_{:}^{:} q(\vec{g}_{:}, \vec{x}_{:}, \vec{v}_{:}, \vec{f}_{:}^{:}, \vec{u}_{:}^{:}) \log \left(\frac{p(\vec{y}_{:}, \vec{g}_{:}, \vec{x}_{:}, \vec{v}_{:}, \vec{f}_{:}^{:}, \vec{u}_{:}^{:})}{q(\vec{g}_{:}, \vec{x}_{:}, \vec{v}_{:}, \vec{f}_{:}^{:}, \vec{u}_{:}^{:})} \right) = \sum_{i=1}^{M} \mathcal{L}_{kin}^{i} + \mathcal{L}_{dyn} \tag{A20}$$

$$= \sum_{i=1}^{M} \sum_{d=1}^{D} \int d\vec{x}_{:}^{i} d\vec{v}_{:}^{i} d(\vec{g}_{:}^{i})_{d} p((\vec{g}_{:}^{i})_{d}|\vec{x}_{:}^{i}, \vec{v}_{:}^{i}) q(\vec{x}_{:}^{i}) q(\vec{v}_{:}^{i}) \log \frac{p((\vec{y}_{:}^{i})_{d}|(\vec{g}_{:}^{i})_{d})}{q(\vec{v}_{:}^{i})} \tag{A21}$$

$$+ \int d\vec{u}_{:}^{:} q(\vec{u}_{:}^{:}) \left[\sum_{t=1}^{T} \int q(\vec{x}_{t}) q(\vec{x}_{-t}) \left(\int d\vec{f}_{t}^{:} \left[\prod_{i=1}^{M} \prod_{j=1}^{M} p(\vec{f}_{t}^{j,i}|\vec{f}_{1:t-1}^{j,i}, \vec{x}_{0:t-1}^{j}, \vec{u}_{:}^{j,i}) \right] \log \prod_{i=1}^{M} p(\vec{x}_{t}^{i}|\vec{f}_{t}^{:,i}, \alpha^{:,i}) \right) \right]$$

$$+ \int d\vec{u}_{:}^{:} q(\vec{u}_{:}^{:}) \log \frac{p(\vec{u}_{:}^{:})}{q(\vec{u}_{:}^{:})} + \sum_{t=0}^{1} \int q(\vec{x}_{t}) \log p(\vec{x}_{t}) d\vec{x}_{t} + H(q(\vec{x}_{:})) \tag{A22}$$

The term in Equation (A21), which we call $\sum_{i=1}^{M} \mathcal{L}_{kin}^{i}$, is the GPLVM ELBO up to $H(q(\vec{x}_{:}))$ and is given in [16]. Next, we consider only the ELBO component which is relevant for the dynamics $\mathcal{L}_{dyn}$ (last two lines of the right hand side of Equation (A22)) and apply the sufficient statistics assumption: knowing $\vec{x}_{-t}^{j}$ and $\vec{u}_{:}^{j,i}$ is sufficient for the $\vec{f}_{t}^{j,i}$ distribution, i.e., $p(\vec{f}_{t}^{j,i}|\vec{f}_{1:t-1}^{j,i}, \vec{x}_{0:t-1}^{j}, \vec{u}_{:}^{j,i}) = p(\vec{f}_{t}^{j,i}|\vec{x}_{0:t-1}^{j}, \vec{u}_{:}^{j,i})$. This assumption lower-bounds $\mathcal{L}_{dyn}$, because it constrains the variational posterior in Equation (A19) away from the correct solution. The sum over the initial latent points in the last line of Equation (A22) may be longer or shorter depending on the dynamics model order, here we use a second order model. The innermost integral can then be written as:

$$\begin{aligned} \mathcal{A} &= \int \left[\prod_{i=1}^{M} \prod_{j=1}^{M} p(\vec{f}_{t}^{j,i}|\vec{x}_{-t}^{j}, \vec{u}_{:}^{j,i}) \right] \log \prod_{i=1}^{M} p(\vec{x}_{t}^{i}|\vec{f}_{t}^{:,i}, \alpha^{:,i}) d\vec{f}_{t}^{:,:} \\ &= \sum_{i=1}^{M} \int \left[\prod_{j=1}^{M} p(\vec{f}_{t}^{j,i}|\vec{x}_{-t}^{j}, \vec{u}_{:}^{j,i}) \right] \log p(\vec{x}_{t}^{i}|\vec{f}_{t}^{:,i}, \alpha^{:,i}) d\vec{f}_{t}^{:,i} \\ &= \sum_{i=1}^{M} \int \left[\prod_{j=1}^{M} \mathcal{N}(\vec{f}_{t}^{j,i}|\vec{\mu}_{\vec{f}_{t}^{j,i}}, \mathbf{S}_{\vec{f}_{t}^{j,i}}) \right] \log \mathcal{N}(\vec{x}_{t}^{i}|\alpha^{i} \sum_{j=1}^{M} (\alpha^{j,i})^{-1} \vec{f}_{t}^{j,i}, \mathbf{I}\alpha^{i}) d\vec{f}_{t}^{:,i} \\ &= \sum_{i=1}^{M} \left[-\tfrac{1}{2} \operatorname{tr} \left[\alpha^{i} \sum_{j=1}^{M} (\alpha^{j,i})^{-2} \mathbf{S}_{\vec{f}_{t}^{j,i}} \right] + \log \mathcal{N} \left(\vec{x}_{t}^{i}|\alpha^{i} \sum_{j=1}^{M} (\alpha^{j,i})^{-1} \vec{\mu}_{\vec{f}_{t}^{j,i}}, \mathbf{I}\alpha^{i} \right) \right] \end{aligned} \tag{A23}$$

$$\vec{\mu}_{\vec{f}_{t}^{j,i}} = \mathbf{K}_{\vec{x}_{-t}^{j}, \vec{z}_{:}^{j,i}} \left(\mathbf{K}_{\vec{z}_{:}^{j,i}, \vec{z}_{:}^{j,i}} \right)^{-1} \vec{u}_{:}^{j,i} \tag{A24}$$

$$\mathbf{S}_{\vec{f}_{t}^{j,i}} = \mathbf{K}_{\vec{x}_{-t}^{j}, \vec{x}_{-t}^{j}} - \mathbf{K}_{\vec{x}_{-t}^{j}, \vec{z}_{:}^{j,i}} \left(\mathbf{K}_{\vec{z}_{:}^{j,i}, \vec{z}_{:}^{j,i}} \right)^{-1} \mathbf{K}_{\vec{x}_{-t}^{j}, \vec{z}_{:}^{j,i}} \tag{A25}$$

$$\alpha^{i} = \left(\sum_{j=1}^{M} (\alpha^{j,i})^{-1} \right)^{-1} \tag{A26}$$

where $\mathbf{K}$ are kernel matrices, obtained by evaluating the kernel function at the pairs of points indicated by the subscripts. Equations (A24) and (A25) follow from the standard formulas for conditional Gaussians (see, e.g., [45], Identities 352–254). Equation (A26) follows from the product-of-experts construction (see Section 3, Equation (1)).

The next step in the evaluation of Equation (A22) is integrating $\mathcal{A}$ over $d\vec{x}_t^{\cdot}$ and $\vec{x}_{-t}^{\cdot}$, which requires averaging kernel matrices. The results of this averaging are denoted by Ψ (see Appendix B and [16] for a derivation). We denote $\Psi_0^{j,i}(\vec{x}_{-t}^{j}) = \int d\vec{x}_{-t}^{j} q(\vec{x}_{-t}^{j}) \mathbf{K}^{j,i}_{\vec{x}_{-t}^{j}, \vec{x}_{-t}^{j}}$, etc.:

$$\begin{aligned}
\mathcal{B} &= \int q(\vec{x}_t^{\cdot}) q(\vec{x}_{-t}^{\cdot})\, \mathcal{A}\, d\vec{x}_t^{\cdot}\, d\vec{x}_{-t}^{\cdot} \\
&= \sum_{i=1}^{M} \Bigg(-\frac{1}{2}\alpha_i \sum_{j=1}^{M} (\alpha^{j,i})^{-2} \operatorname{tr}\left[\Psi_0^{j,i}(\vec{x}_{-t}^{j}) - \left(\mathbf{K}^{j,i}_{\vec{z}_{:}^{j,i}, \vec{z}_{:}^{j,i}} \right)^{-1} \Psi_2^{j,i}(\vec{x}_{-t}^{j}) \right] \\
&- Q_i \log \sqrt{2\pi\alpha^i} - \frac{1}{2}(\alpha^i)^{-1} \left[\operatorname{tr}(\mathbf{S}_{\vec{x}_t^i}) + \vec{\mu}_{\vec{x}_t^i}^T \vec{\mu}_{\vec{x}_t^i} \right] \\
&+ \vec{\mu}_{\vec{x}_t^i}^T \Big(\sum_{j=1}^{M} (\alpha^{j,i})^{-1} \Psi_1^{j,i}(\vec{x}_{-t}^{j}) \left(\mathbf{K}^{j,i}_{\vec{z}_{:}^{j,i}, \vec{z}_{:}^{j,i}} \right)^{-1} \vec{u}_{:}^{j,i} \Big) \\
&- \frac{1}{2}\alpha^i \sum_{j=1}^{M} \sum_{k=1}^{M} (\alpha^{j,i})^{-1} (\alpha^{k,i})^{-1} \vec{u}_{:}^{j,i\,T} \left(\mathbf{K}^{j,i}_{\vec{z}_{:}^{j,i}, \vec{z}_{:}^{j,i}} \right)^{-1} \Psi_2^{j,k,i}(\vec{x}_{-t}^{j}, \vec{x}_{-t}^{k}) \left(\mathbf{K}^{k,i}_{\vec{z}_{:}^{k,i}, \vec{z}_{:}^{k,i}} \right)^{-1} \vec{u}_{:}^{k,i} \Bigg)
\end{aligned} \tag{A27}$$

Next, we sum this expression over time points:

$$\begin{aligned}
\mathcal{C} &= \sum_{t=1}^{T} \mathcal{B} \\
&= \sum_{i=1}^{M} \Bigg(\sum_{t=1}^{T} \Bigg[-\frac{1}{2}\alpha^i \sum_{j=1}^{M} (\alpha^{j,i})^{-2} \operatorname{tr}\left[\Psi_0^{j,i}(\vec{x}_{-t}^{j}) - \left(\mathbf{K}^{j,i}_{\vec{z}_{:}^{j,i}, \vec{z}_{:}^{j,i}} \right)^{-1} \Psi_2^{j,i}(\vec{x}_{-t}^{j}) \right] \\
&- \log \sqrt{2\pi\alpha^i} - \frac{1}{2}(\alpha^i)^{-1} \left[\operatorname{tr}(\mathbf{S}_{\vec{x}_t^i}) + \vec{\mu}_{\vec{x}_t^i}^T \vec{\mu}_{\vec{x}_t^i} \right] \Bigg] \\
&+ \sum_{j=1}^{M} (\alpha^{j,i})^{-1} \left[\sum_{t=1}^{T} \vec{\mu}_{\vec{x}_t^i}^T \Psi_1^{j,i}(\vec{x}_{-t}^{j}) \right] \left(\mathbf{K}^{j,i}_{\vec{z}_{:}^{j,i}, \vec{z}_{:}^{j,i}} \right)^{-1} \vec{u}_{:}^{j,i} \\
&- \frac{1}{2}\alpha^i \sum_{j=1}^{M} \sum_{k=1}^{M} (\alpha^{j,i})^{-1} (\alpha^{k,i})^{-1} \vec{u}_{:}^{j,i\,T} \left(\mathbf{K}^{j,i}_{\vec{z}_{:}^{j,i}, \vec{z}_{:}^{j,i}} \right)^{-1} \left[\sum_{t=1}^{T} \Psi_2^{j,k,i}(\vec{x}_{-t}^{j}, \vec{x}_{-t}^{k}) \right] \left(\mathbf{K}^{j,i}_{\vec{z}_{:}^{k,i}, \vec{z}_{:}^{k,i}} \right)^{-1} \vec{u}_{:}^{k,i} \Bigg)
\end{aligned} \tag{A28}$$

For every part $i \in 1 \ldots M$, we stack up the $\vec{u}_{:}^{:,i}$ into $\vec{u}^i$ (first by IV-index $k = 1, \ldots, K$, and then by part index such that $\vec{u}_k^{j,i} = (\vec{u}^i)_{Q^i \cdot (K \cdot j + k) + q^i + 1}$) and construct a large block matrices $\mathcal{F}^i$ and stacked vector $\mathcal{G}^i$ with block elements

$$\mathcal{F}^i_{j,k} = \alpha^i \alpha_{j,i}^{-1} \alpha_{k,i}^{-1} \left(\mathbf{K}^{j,i}_{\vec{z}_{:}^{j,i}, \vec{z}_{:}^{j,i}} \right)^{-1} \left[\sum_{t=1}^{T} \Psi_2^{j,k,i}(\vec{x}_{-t}^{j}, \vec{x}_{-t}^{k}) \right] \left(\mathbf{K}^{k,i}_{\vec{z}_{:}^{k,i}, \vec{z}_{:}^{k,i}} \right)^{-1} \tag{A29}$$

$$\mathcal{G}^i_j = \left(\alpha_{j,i}^{-1} \left[\sum_{t=1}^{T} \vec{\mu}_{\vec{x}_t^i}^T \Psi_1^{j,i}(\vec{x}_{-t}^{j}) \right] \left(\mathbf{K}^{j,i}_{\vec{z}_{:}^{j,i}, \vec{z}_{:}^{j,i}} \right)^{-1} \right)^T \tag{A30}$$

For $j \neq k$: $\Psi_2^{j,k,i}(\vec{x}_{-t}^{j}, \vec{x}_{-t}^{k}) = \Psi_1^{j,i}(\vec{x}_{-t}^{j}) \Psi_1^{k,i}(\vec{x}_{-t}^{k})$. Otherwise, $\Psi_2^{j,j,i}(\vec{x}_{-t}^{j}, \vec{x}_{-t}^{j}) = \Psi_2^{j,i}(\vec{x}_{-t}^{j})$. We rewrite $\mathcal{C}$ as a quadratic form in the stacked augmenting IVs $\vec{u}^i$ to facilitate closed-form optimization of the dynamics ELB in Equation (A22) with respect to the stacked IV density $q(\vec{u}^{\cdot})$ using variational calculus, as described in Appendix A:

$$\begin{aligned}\mathcal{C} &= \sum_{i=1}^{M}\left[-\frac{1}{2}\vec{u}^{iT}\mathcal{F}^{i}\vec{u}^{i}+\vec{u}^{iT}\mathcal{G}^{i}+\mathcal{H}^{i}\right]\\ &= \sum_{i=1}^{M}\left[-\frac{1}{2}(\vec{u}^{i}-\mathcal{F}^{i-1}\mathcal{G}^{i})^{T}\mathcal{F}^{i}(\vec{u}^{i}-\mathcal{F}^{i-1}\mathcal{G}^{i})+\frac{1}{2}\mathcal{G}^{iT}\mathcal{F}^{i-1}\mathcal{G}^{i}+\mathcal{H}^{i}\right] \qquad \text{(A31)}\\ \mathcal{C} &= \sum_{i=1}^{M}\mathcal{C}^{i} \qquad \text{(A32)}\\ \mathcal{C}^{i} &= -\frac{1}{2}(\vec{u}^{i}-\mathcal{F}^{i-1}\mathcal{G}^{i})^{T}\mathcal{F}^{i}(\vec{u}^{i}-\mathcal{F}^{i-1}\mathcal{G}^{i})+\frac{1}{2}\mathcal{G}^{iT}\mathcal{F}^{i-1}\mathcal{G}^{i}+\mathcal{H}^{i} \qquad \text{(A33)}\\ \mathcal{H}^{i} &= \sum_{t=1}^{T}\left[-\frac{1}{2}\alpha_{i}\sum_{j=1}^{M}(\alpha_{j,i}^{-1})^{2}\,\mathrm{tr}\left[\Psi_{0}^{j,i}(\vec{x}_{-t}^{j})-\left(\mathbf{K}_{\vec{z}_{:}^{j,i},\vec{z}_{:}^{j,i}}^{j,i}\right)^{-1}\Psi_{2}^{j,i}(\vec{x}_{-t}^{j})\right]\right.\\ &\quad\left.-\log\sqrt{2\pi\alpha^{i}}-\frac{1}{2}\alpha_{i}^{-1}\left[\mathrm{tr}(\mathbf{S}_{\vec{x}_{t}^{i}})+\vec{\mu}_{\vec{x}_{t}^{i}}^{T}\vec{\mu}_{\vec{x}_{t}^{i}}\right]\right] \qquad \text{(A34)}\end{aligned}$$

After this optimization, we can write the dynamics ELBO using $p(\vec{u}^{i}) = \prod_{j=1}^{M} p(\vec{u}_{:}^{j,i}) = \prod_{j=1}^{M}\mathcal{N}(\vec{u}^{j,i}|0,\mathbf{K}_{\vec{z}_{:}^{j,i},\vec{z}_{:}^{j,i}}) = \mathcal{N}(\vec{u}^{i}|0,\mathbf{K}_{\vec{z}_{:}^{:,i},\vec{z}_{:}^{:,i}})$ where $\mathbf{K}_{\vec{z}_{:}^{:,i},\vec{z}_{:}^{:,i}}$ is a block-diagonal covariance matrix with the blocks given by the individual $\mathbf{K}_{\vec{z}_{:}^{j,i},\vec{z}_{:}^{j,i}}$:

$$\begin{aligned}\mathcal{L}_{dyn}(\Theta) &\geq \log\int p(\vec{u}_{:}^{:})\exp(\mathcal{C})d\vec{u}_{:}^{:}+H(q(\vec{x}_{:}^{:}))\\ &= \log\prod_{i=1}^{M}\int p(\vec{u}^{i})\exp(\mathcal{C}^{i})d\vec{u}^{i}+H(q(\vec{x}_{:}^{:}))\\ &= \sum_{i=1}^{M}\left[\log\int p(\vec{u}^{i})\exp(-\frac{1}{2}(\vec{u}^{i}-\mathcal{F}^{i-1}\mathcal{G}^{i})^{T}\mathcal{F}^{i}(\vec{u}^{i}-\mathcal{F}^{i-1}\mathcal{G}^{i}))d\vec{u}^{i}+\frac{1}{2}\mathcal{G}^{iT}\mathcal{F}^{i-1}\mathcal{G}^{i}+\mathcal{H}^{i}\right]+H(q(\vec{x}_{:}^{:})) \qquad \text{(A35)}\\ &= \sum_{i=1}^{M}\left[-\log\sqrt{(2\pi)^{\dim(\mathcal{F}^{i-1})}|\mathcal{F}^{i-1}+\mathbf{K}_{\vec{z}_{:}^{:,i},\vec{z}_{:}^{:,i}}|}-\frac{1}{2}\mathcal{G}^{iT}\mathcal{F}^{i-1}(\mathcal{F}^{i-1}+\mathbf{K}_{\vec{z}_{:}^{:,i},\vec{z}_{:}^{:,i}})^{-1}\mathcal{F}^{i-1}\mathcal{G}^{i}+\log Z(\mathcal{F}^{i-1})\right]\\ &+\sum_{i=1}^{M}\left[\frac{1}{2}\mathcal{G}^{iT}\mathcal{F}^{i-1}\mathcal{G}^{i}+\mathcal{H}^{i}\right]+H(q(\vec{x}_{:}^{:}))\end{aligned}$$

This is the expression which we optimize with respect to $q(\vec{x}_{:}^{:})$ and $q(\vec{u}_{:}^{:})$. Since the stacked dynamics IVs $\vec{u}^{i}$ do not interact across parts in this expression (Line 2), it follows that density $q(\vec{u}_{:}^{:})$ factorizes across parts. Their optimal density for each part is given by (cf. Equation (A6), Z is the normalization constant of the multivariate Gaussian):

$$\begin{aligned}q(\vec{u}^{i}) &= \frac{1}{Z}p(\vec{u}^{i})\exp(\mathcal{C}^{i})\\ &= \frac{1}{Z}\mathcal{N}(\vec{u}^{i}|0,\mathbf{K}_{\vec{z}_{:}^{:,i},\vec{z}_{:}^{:,i}})\exp(-\frac{1}{2}(\vec{u}^{i}-\mathcal{F}^{i-1}\mathcal{G}^{i})^{T}\mathcal{F}^{i}(\vec{u}^{i}-\mathcal{F}^{i-1}\mathcal{G}^{i})) \qquad \text{(A36)}\\ &= \mathcal{N}(\vec{u}^{i}|(\mathbf{K}_{\vec{z}_{:}^{:,i},\vec{z}_{:}^{:,i}}^{-1}+\mathcal{F}^{i})^{-1}\mathcal{G}^{i},(\mathbf{K}_{\vec{z}_{:}^{:,i},\vec{z}_{:}^{:,i}}^{-1}+\mathcal{F}^{i})^{-1})\end{aligned}$$

References

1. Bizzi, E.; Cheung, V.; d'Avella, A.; Saltiel, P.; Tresch, M. Combining modules for movement. *Brain Res. Rev.* **2008**, *57*, 125–133. [CrossRef] [PubMed]
2. Endres, D.; Chiovetto, E.; Giese, M. Model selection for the extraction of movement primitives. *Front. Comput. Neurosci.* **2013**, *7*, 185. [CrossRef] [PubMed]
3. Schaal, S. Dynamic Movement Primitives -A Framework for Motor Control in Humans and Humanoid Robotics. In *Adaptive Motion of Animals and Machines*; Kimura, H., Tsuchiya, K., Ishiguro, A., Witte, H., Eds.; Springer: Tokyo, Japan, 2006; pp. 261–280. [CrossRef]

4. Giese, M.A.; Mukovskiy, A.; Park, A.N.; Omlor, L.; Slotine, J.J.E. Real-Time Synthesis of Body Movements Based on Learned Primitives. In *Statistical and Geometrical Approaches to Visual Motion Analysis*; Cremers, D., Rosenhahn, B., Yuille, A.L., Eds.; Springer: Berlin/Heidelberg, Germany, 2009; Volume 5604, pp. 107–127.
5. Koch, K.H.; Clever, D.; Mombaur, K.; Endres, D.M. Learning Movement Primitives from Optimal and Dynamically Feasible Trajectories for Humanoid Walking. In Proceedings of the IEEE/RAS International Conference on Humanoid Robots (Humanoids 2015), Seoul, Korea, 3–5 November 2015; pp. 866–873.
6. Chiovetto, E.; Berret, B.; Pozzo, T. Tri-dimensional and triphasic muscle organization of whole-body pointing movements. *Neuroscience* **2010**, *170*, 1223–1238. [CrossRef] [PubMed]
7. Omlor, L.; Giese, M.A. Anechoic Blind Source Separation using Wigner Marginals. *J. Mach. Learn. Res.* **2011**, *12*, 1111–1148.
8. Chiovetto, E.; Giese, M.A. Kinematics of the coordination of pointing during locomotion. *PLoS ONE* **2013**, *8*, e79555. [CrossRef] [PubMed]
9. Clever, D.; Harant, M.; Koch, K.H.; Mombaur, K.; Endres, D.M. A novel approach for the generation of complex humanoid walking sequences based on a combination of optimal control and learning of movement primitives. *Robot. Autom. Syst.* **2016**, *83*, 287–298. [CrossRef]
10. Mussa-Ivaldi, F.A.; Solla, S.A. Neural Primitives for Motion Control. *IEEE J. Ocean. Eng.* **2004**, *29*, 640–650. [CrossRef]
11. Hart, C.B.; Giszter, S. Distinguishing Synchronous and Time Varying Synergies using Point Process Interval Statistics: Motor Primitives in Frog and Rat. *Front. Comput. Neurosci.* **2013**, *7*, 52. [CrossRef] [PubMed]
12. Ijspeert, A.J.; Nakanishi, J.; Hoffmann, H.; Pastor, P.; Schaal, S. Dynamical Movement Primitives: Learning Attractor Models for Motor Behaviors. *Neural Comput.* **2013**, *25*, 328–373. [CrossRef] [PubMed]
13. Rückert, E.; d'Avella, A. Learned Parameterized Dynamic Movement Primitives with Shared Synergies for Controlling Robotic and Musculoskeletal Systems. *Front. Comput. Neurosci.* **2013**, *7*, 138. [CrossRef] [PubMed]
14. Velychko, D.; Endres, D.; Taubert, N.; Giese, M.A. Coupling Gaussian Process Dynamical Models with Product-of-Experts Kernels. In *International Conference on Artificial Neural Networks*; Springer: Cham, Switzerland, 2014; pp. 603–610. [CrossRef]
15. Rasmussen, C.E.; Williams, C.K.I. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*; The MIT Press: Cambridge, MA, USA, 2005.
16. Titsias, M.K.; Lawrence, N.D. Bayesian Gaussian Process Latent Variable Model. In Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS), Sardinia, Italy, 13–15 May 2010; pp. 844–851.
17. Bizzi, E.; Cheung, V.C. The Neural Origin of Muscle Synergies. *Front. Comput. Neurosci.* **2013**, *7*, 51. [CrossRef] [PubMed]
18. Földiák, P.; Endres, D. Sparse coding. *Scholarpedia* **2008**, *3*, 2984. [CrossRef]
19. Velychko, D.; Knopp, B.; Endres, D. The variational coupled Gaussian Process Dynamical Model. In *International Conference on Artificial Neural Networks*; Springer: Cham, Switzerland, 2017; pp. 291–299. [CrossRef]
20. Candela, J.Q.; Rasmussen, C.E. A Unifying View of Sparse Approximate Gaussian Process Regression. *J. Mach. Learn. Res.* **2005**, *6*, 1939–1959.
21. Snelson, E.; Ghahramani, Z. Sparse Gaussian processes using pseudo-inputs. In *Advances in Neural Information Processing Systems 18*; MIT Press: Cambridge, MA, USA, 2006; pp. 1257–1264.
22. Lawrence, N.D. Learning for Larger Datasets with the Gaussian Process Latent Variable Model. *Artif. Intell. Stat.* **2007**, *2*, 243–250.
23. Titsias, M.K. Variational Learning of Inducing Variables in Sparse Gaussian Processes. *Artif. Intell. Stat.* **2009**, *5*, 567–574.
24. Damianou, A.C.; Titsias, M.; Lawrence, N.D. Variational Gaussian Process Dynamical Systems. In *Advances in Neural Information Processing Systems 24*; Shawe-Taylor, J., Zemel, R., Bartlett, P., Pereira, F., Weinberger, K., Eds.; MIT Press: Cambridge, MA, USA, 2011; pp. 2510–2518.
25. Wang, J.M.; Fleet, D.J.; Hertzmann, A. Gaussian Process Dynamical Models for Human Motion. *IEEE Trans. Pattern Anal. Mach. Intell.* **2008**, *30*, 283–298. [CrossRef] [PubMed]
26. Urtasun, R.; Fleet, D.J.; Lawrence, N.D. Modeling Human Locomotion with Topologically Constrained Latent Variable Models. In *Human Motion–Understanding, Modeling, Capture and Animation*; Springer: Berlin/Heidelberg, Germany, 2007; Volume 4814, pp. 104–118.

27. Taubert, N.; Endres, D.; Christensen, A.; Giese, M.A. Shaking Hands in Latent Space. In *Annual Conference on Artificial Intelligence*; Springer: Berlin/Heidelberg, Germany, 2011; Volume 7006, pp. 330–334.
28. Levine, S.; Wang, J.M.; Haraux, A.; Popović, Z.; Koltun, V. Continuous Character Control with Low-Dimensional Embeddings. *ACM Trans. Graph.* **2012**, *31*, 28. [CrossRef]
29. Chen, J.; Kim, M.; Wang, Y.; Ji, Q. Switching Gaussian Process Dynamic Models for simultaneous composite motion tracking and recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2009 (CVPR 2009), Miami, FL, USA, 20–25 June 2009; pp. 2655–2662. [CrossRef]
30. Mattos, C.L.C.; Dai, Z.; Damianou, A.; Forth, J.; Barreto, G.A.; Lawrence, N.D. Recurrent Gaussian Processes. *arXiv* **2015**, arxiv:1511.06644.
31. Frigola, R.; Lindsten, F.; Schön, T.B.; Rasmussen, C. Bayesian Inference and Learning in Gaussian Process State-Space Models with Particle MCMC. In *Advances in Neural Information Processing Systems 26*; Burges, C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2013; pp. 3156–3164.
32. Frigola, R.; Chen, Y.; Rasmussen, C. Variational Gaussian Process State-Space Models. In *Advances in NIPS 27*; Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., Weinberger, K., Eds.; NIPS Foundation: Montreal, QC, Canada, 2014; pp. 3680–3688.
33. Bauer, M.; van der Wilk, M.; Rasmussen, C. Understanding Probabilistic Sparse Gaussian Process Approximations. *arXiv* **2016**, arxiv:1606.04820.
34. Taubert, N.; Christensen, A.; Endres, D.; Giese, M. Online simulation of emotional interactive behaviors with hierarchical Gaussian process dynamical models. In Proceedings of the ACM Symposium on Applied Perception (SAP '12), Los Angeles, CA, USA, 3–4 August 2012; pp. 25–32. [CrossRef]
35. Hinton, G.E. Products of Experts. In Proceedings of the 9th International Conference on Artificial Neural Networks (ICANN'99), Edinburgh, UK, 7–10 September 1999; Volume 1, pp. 1–6.
36. Bishop, C.M. *Pattern Recognition and Machine Learning (Information Science and Statistics)*; Springer: Secaucus, NJ, USA, 2006.
37. Bastien, F.; Lamblin, P.; Pascanu, R.; Bergstra, J.; Goodfellow, I.J.; Bergeron, A.; Bouchard, N.; Bengio, Y. Theano: New features and speed improvements. *arXiv* **2012**, arxiv:1211.5590.
38. Jones, E.; Oliphant, T.; Peterson, P. SciPy: Open Source Scientific Tools for Python. Available online: https://www.scipy.org/ (accessed on 9 October 2015).
39. Sakoe, H.; Chiba, S. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Trans. Acoust. Speech Signal Process.* **1978**, *26*, 43–49. [CrossRef]
40. Taubert, N.; Löffler, M.; Ludolph, N.; Christensen, A.; Endres, D.; Giese, M. A virtual reality setup for controllable, stylized real-time interactions between humans and avatars with sparse Gaussian process dynamical models. In Proceedings of the ACM Symposium on Applied Perception (SAP '13), Dublin, Ireland, 22–23 August 2013; pp. 41–44. [CrossRef]
41. Paraschos, A.; Daniel, C.; Peters, J.; Neumann, G. Probabilistic Movement Primitives. In *Advances in NIPS 26*; Burges, C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K., Eds.; NIPS Foundation: Montreal, QC, Canada, 2013; pp. 2616–2624.
42. Pastor, P.; Kalakrishnan, M.; Righetti, L.; Schaal, S. Towards Associative Skill Memories. In Proceedings of the 2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids), Osaka, Japan, 29 November–1 December 2012; pp. 309–315.
43. Land, W.M.; Rosenbaum, D.A.; Seegelke, C.; Schack, T. Whole-body posture planning in anticipation of a manual prehension task: Prospective and retrospective effects. *Acta Psychol.* **2013**, *144*, 298–307. [CrossRef] [PubMed]
44. Deisenroth, M.; Fox, D.; Rasmussen, C. Gaussian Processes for Data-Efficient Learning in Robotics and Control. *Pattern Anal. Mach. Intell. IEEE Trans.* **2015**, *37*, 408–423. [CrossRef] [PubMed]
45. Petersen, K.B.; Pedersen, M.S. *The Matrix Cookbook*; Version 20121115; Technical University of Denmark: Lyngby, Denmark, 2012.

Predicting Perceived Naturalness of Human Animations Based on Generative Movement Primitive Models

BENJAMIN KNOPP, DMYTRO VELYCHKO, JOHANNES DREIBRODT, and DOMINIK ENDRES, University of Marburg

15

We compared the perceptual validity of human avatar walking animations driven by six different representations of human movement using a graphics Turing test. All six representations are based on movement primitives (MPs), which are predictive models of full-body movement that differ in their complexity and prediction mechanism. Assuming that humans are experts at perceiving biological movement from noisy sensory signals, it follows that these percepts should be describable by a suitably constructed Bayesian ideal observer model. We build such models from MPs and investigate if the perceived naturalness of human animations are predictable from approximate Bayesian model scores of the MPs. We found that certain MP-based representations are capable of producing movements that are perceptually indistinguishable from natural movements. Furthermore, approximate Bayesian model scores of these representations can be used to predict perceived naturalness. In particular, we could show that movement dynamics are more important for perceived naturalness of human animations than single frame poses. This indicates that perception of human animations is highly sensitive to their temporal coherence. More generally, our results add evidence for a shared MP-representation of action and perception. Even though the motivation of our work is primarily drawn from neuroscience, we expect that our results will be applicable in virtual and augmented reality settings, when perceptually plausible human avatar movements are required.

CCS Concepts: • **Computing methodologies** → **Perception**; *Animation*; *Motion processing*; • **Theory of computation** → *Gaussian processes*;

Additional Key Words and Phrases: Human animation, movement primitives, perception, dynamical systems, psychophysics, Gaussian process dynamical model, dynamical movement primitives

ACM Reference format:
Benjamin Knopp, Dmytro Velychko, Johannes Dreibrodt, and Dominik Endres. 2019. Predicting Perceived Naturalness of Human Animations Based on Generative Movement Primitive Models. *ACM Trans. Appl. Percept.* 16, 3, Article 15 (September 2019), 18 pages.
https://doi.org/10.1145/3355401

This work was funded by DFG, IRTG1901 - The brain in action, and SFB-TRR 135 - Cardinal mechanisms of perception.
Authors' addresses: B. Knopp, D. Velychko, J. Dreibrodt, and D. Endres, Department of Psychology, University of Marburg, Gutenbergstraße 18, 35039 Marburg; emails: {benjamin.knopp, dmytro.velychko}@uni-marburg.de, dreibrod@students.uni-marburg.de, dominik.endres@uni-marburg.de.

1544-3558/2019/09-ART15
https://doi.org/10.1145/3355401

1 INTRODUCTION

The perception of biological movement[1] is of paramount importance for humans: in many situations, in real life as well as in virtual reality, it is necessary to predict internal states and goals of other actors from observed body movements. Such predictions are facilitated by a model of relevant degrees of freedom (DOF), and the abstraction of redundant ones. Strong evidence for the existence of such a model from a neuroscientific perspective is provided by the point-light walker experiments of Johansson (1994): just a few dots resembling the human body's spatial configuration and dynamics are enough for robust detection of activities like walking, dancing, and the like. Practical evidence is given by the everlasting struggle of animators to produce perceptually valid human animations (without relying on motion captured data).

A related abstraction problem must be solved in motor production: our bodies have many more DOFs than needed for any given movement (Bernstein 1967); hence, the redundant DOFs need to be bound or remain uncontrolled. One way to bind these DOFs is via *movement primitives* (MPs) or synergies, as predicted by optimal control feedback theory (Todorov and Jordan 2003).

This relationship between movement perception and production suggests that a shared representation might be used to address them both, as proposed by the *common coding* hypothesis and the theory of event coding (Friston 2010; Hommel et al. 2001; Prinz 1997; Shin et al. 2010; Wolpert et al. 2003). However, this hypothesis does not specify the level of representation on which the common coding happens. We therefore investigate whether MPs are candidates for such a shared representation. Their suitability for complex movement production has already been demonstrated (Clever et al. 2017; Giszter 2015; Ijspeert et al. 2013; Omlor and Giese 2011), we would like to determine how close human perceptual performance is to an "ideal observer" comprised of MPs.

The "ideal observer" assumption is motivated by the apparent ease with which we perceive and interpret our fellow humans' movements: we hypothesize that movement perception is another instance where we behave nearly Bayes-optimally (Knill and Pouget 2004). Hence, human perceptual expectations should be predictable by Bayesian model comparison between MP models. To test this hypothesis, we trained generative MP models on kinematic data of walking movements, and compared movements based on these MPs in a Graphics Turing Test. We are also interested in determining the model scores which are most predictive of human expectations.

2 RELATED WORK

Biological motion perception induced by point-light-stimuli is a related, and heavily investigated research topic (for an overview, see Troje (2013)): point-light stimuli, first introduced to demonstrate the perceptual binding of different points to one "Gestalt" (Johansson 1994), they have been used to study the perception of movement isolated from body shape and other cues (Bertenthal and Pinto 1994; Casile and Giese 2005; Troje 2002; Troje et al. 2005).

We are not concerned with the shape inference process from point-light-displays or stick figures, therefore we use 3D avatars, which are closer to natural stimuli. It has been shown that human observers have a higher sensitivity for detecting differences in movement when using 3D avatars compared to stick figures (Hodgins et al. 1998).

Motivation to use MPs as perceptual representations of movement is given by an action-perception coupling on the neural level (Dayan et al. 2007): the famous "2/3 power law", an obseved invariant in curved drawing movements, seems to have a perceptual representation in the brain. Parabolic MPs can simultaneously obey the 2/3 power law and minimize jerk, which has been proposed as a control principle for arm movements (Polyakov et al. 2009). Perceptual experiments investigating the segmentation of taekwondo solo forms imply that higher order polynomial MPs might be more appropriate perceptual descriptors for full-body movement (Endres et al. 2011).

[1]The term "biological motion" has been used to denote a point-light display of (biological) movement. We use the term 'human animation' for a 3D-rendered display of movement.

In an experiment similar to ours, it has been shown that hierarchical Gaussian process dynamical models can synthesize hand shake movements indistinguishable from natural ones (Taubert et al. 2012). Furthermore, the perception of emotion based on spatio-temporal MPs has been investigated by Roether et al. (2009) and Chiovetto et al. (2018). In our study, we are interested in comparing different MP types in a unified Bayesian framework (Endres et al. 2013) with respect to the perception of naturalness.

3 MODELS AND EXPERIMENTAL METHODS

In this section, we first introduce the investigated MP models, which are used to generate the stimuli for graphics Turing test (McGuigan 2006). Next, we describe our experiment designed to determine the perceived naturalness of the generated walking movements. Finally, we explain the data analysis methods used to predict the perceived naturalness from approximate Bayesian model scores.

3.1 Movement Primitives

MPs refer to building blocks of complex movements, but there is little consensus on an exact definition. Consequently, many different types of MPs have been proposed in literature (Endres et al. 2013). These types can be classified as spatial (Giszter et al. 1992; Tresch et al. 1999), temporal (Clever et al. 2016; Endres et al. 2013), spatio-temporal (d'Avella et al. 2003; Omlor and Giese 2011) and dynamical MPs (Ijspeert et al. 2013).

We focus on dynamical and temporal MPs in this study, as we are interested in finding a higher level representation suitable for modeling perception, as opposed to spatial MPs, which have been used to model muscle synergies in the spinal chord (Giszter 2015). Anechoic mixture models have been proposed to enable phase shifted combinations of MPs (Chiovetto et al. 2018; Omlor and Giese 2011). We do not explicitly test this type of MP here, since the relative phase shifts the walking movements we studied are negligbile.

We perceptually validate 6 generative MP models: Temporal MPs, Dynamical MPs and 4 flavors of the Gaussian Process Dynamical Model (GPDM) (Velychko et al. 2018; Wang et al. 2008): GPDM, variational GPDM, coupled GPDM, and variational coupled GPDM.

In this section, we can only provide a rough overview, just enough to enable readers from different backgrounds to understand parameters of the stimuli for the psychophysical experiment. Please refer to the cited papers for detailed information. Velychko et al. (2018) also provide graphical model representations and summarize the features of the MP models presented in this chapter.

3.1.1 Temporal Movement Primitives (TMP) (Clever et al. 2016). Temporal MPs describe the stereotyped temporal patterns of movement parameters (for example EMG, but also joint trajectories as well as endpoint trajectories). A possible biological implementation of temporal MPs might be central pattern generators (CPGs) (Ivanenko et al. 2004) combined with cortical top-down control. Temporal MPs incorporate a temporal predictive mechanism: the complete time-course of the movement is determined at its onset. This type of MPs allows for simple concatenation and temporal scaling.

The trajectory $x_k(t)$ of a DOF X_k, e.g., a joint angle, is a weighted sum of Q MPs Y_q, which are functions of time $y_q(t)$. $\varepsilon_i(t) \sim \mathcal{N}(0, \sigma_i)$ is Gaussian observation noise:

$$x_k(t) = \sum_{q=1}^{Q} w_{k,q} y_q(t) + \varepsilon_i(t). \tag{1}$$

We treat the number of MPs as ideal observer model parameter to be determined. In general, more MPs allow for more fine-grained temporal structure of the movement, but might lead to over-fitting. To determine the MPs and their number, we follow the approach of Clever et al. (2016): weights w and MPs Y_q have a Gaussian Process (GP) prior and are learned from the training data by maximizing a variational lower bound on the Bayesian model evidence (ELBO, evidence lower bound). The ELBO is equal to the negative free energy (Friston 2010). In

keeping with the free energy/Bayesian brain theory, one would therefore expect that the ELBO should be useful for selecting the appropriate number of MPs Q for the generation of perceptually valid movements.

3.1.2 Dynamic Movement Primitives (DMP) (Ijspeert et al. 2013). While temporal MPs directly model the movement parameters (e.g., trajectories or muscle activations), DMPs describe the stereotyped elements of movement as attractors of a dynamical system, thus enabling the prediction of the next state from the previous ones. Building on the hypothesis of separate brain areas for rhythmic and discrete movements, two kinds of dynamical systems are common: cyclic oscillators and point attractors (Schaal 2006).

More formally: DMP models represent a movement trajectory $x_k(t)$ obeying a differential equation. They rely on a damped spring system which forces $x_k(t)$ to contract to the specified goal g_k, if the dampening factor is high enough. Through the non-linear forcing function f_k (Equation (2)) the trajectories can be modified. This function is modeled as weighted sum of Gaussian basis functions $\Psi_i(\tau)$ (Equation (4)). Time is replaced by τ, which decays exponentially to zero (Equation (3)). DMPs are learned from training data by setting the weights w_i such that the training mean-squared error (MSE) is minimal.

$$\tau \ddot{x}_k = \alpha_z(\beta_z(g_k - x_k) - \dot{x}_k) + f_k(\tau) \tag{2}$$

$$\dot{\tau} \propto -\tau \tag{3}$$

$$f_k(\tau) = \frac{\sum_{i=1}^{N} \Psi_i(\tau) w_{k,i}}{\sum_{i=1}^{N} \Psi_i(\tau)} \tau (g_k - x_k(0)). \tag{4}$$

The number of basis functions N is the ideal observer model complexity parameter. It serves a similar role as the number of MPs in the TMP model: more basis functions allow for more complicated forcing functions, which enable richer temporal dynamics. The number can, e.g., be selected by cross-validation, we investigate if N reflects the perceived naturalness.

3.1.3 Gaussian Process Dynamical Model (GPDM) (Wang et al. 2008). Learnable dynamical systems for movement representation have been proposed in the context of computer graphics: the GPDM is a state-space model, which learns a dynamical mapping in a latent space of the whole-body movement. Such a model is also physiologically attractive, because it is able to reflect the dynamic nature of the environment and the body itself, without explicit assumptions of their form (Shenoy et al. 2013; Sussillo et al. 2015).

In contrast to DMPs, GPDMs learn a full dynamical model of latent variables Y in discrete time, which are mapped onto the observed DOFs X_k. Both the dynamics mapping $f()$ (Equation (5)), as well as the mapping from latent to observed space $g()$ (Equation (6)) are drawn from Gaussian process priors, hence the name. dt denotes the time discretization step-size:

$$y(t) = f(y(t - dt)) + \varepsilon_{y,t}, \tag{5}$$

$$x_k(t) = g_k(y(t)) + \varepsilon_{x,t}. \tag{6}$$

There are two main drawbacks which make the GPDM unlikely as a perceptual MP model: (1) there is no (obvious) way of a recombination operation that would make GPDMs modular. Modularity here refers to the possibility of generating a large repertoire of movements from the recombination of a small number of MPs. (2) Due to the non-parametric GPs prior, the movements *are* the movement representation, which is not compact.

A further consequence of this non-parametric prior is no explicit ideal observer model complexity parameter. Therefore, we compare the GPDM estimated by maximum *a-posteriori* inference (MAP) with the other movement primitive representations. The GPDM can also be trained by variational inference, giving rise to the vGPDM. This is a special case of the variational coupled GPDM described in 3.1.5.

3.1.4 Coupled Gaussian Process Dynamical Model (cGPDM) (Velychko et al. 2014). The cGPDM was proposed to make GPDMs modular. Here, one learns different dynamical models for different body parts. Each body part is described by a GPDM, where the latent variables predict not only the next time-step of their associated body part,

but also the temporal evolution of other body parts via coupling functions. This way, flexible coupling between body parts is possible. The vCGPDM can be regarded as a middle ground between DMPs encoding single DOFs, and the monolithic GPDM. The latent dynamical systems can thus be thought of as flexibly coupled CPGs routing commands to the muscles.

As with the MAP-trained GPDM introduced in the previous section, there is no explicit ideal observer model complexity parameter in the MAP-trained cGPDM.

3.1.5 Variational (Coupled) Gaussian Process Dynamical Model (v(C)GPDM) (Velychko et al. 2018). The vCGPDM compresses the movement representation of cGPDMs by introducing sparse variational approximations with a deterministic learning scheme. Here, each MP is parameterized by a small set of inducing points (IPs) and associated inducing values (IVs), leading to a compact representation with constant storage requirements. Flexible recombination of these IPs/IVs for each body part enables the required modularity. The initial choice of IPs/IVs is the only remaining source of stochasticity in the training process. It may have measurable effects, as we will show below. We use IPs for both mappings, seving as ideal observer model parameters: "dynamics" IPs for the dynamical model mapping, and "pose" IPs for the latent-to-observed variable mapping. More dynamics IPs allow for richer dynamics (similar to the parameters of DMP and TMP), while more pose IPs will allow for more (spatial) variability of poses.

An IP/IV pair might be thought of as a prototypical example for the mappings drawn from their associated Gaussian process. They thus provide some abstraction from the observed movement and might be implemented by small neuronal populations. Similar to the TMP, the vCGPDM is trained by maximizing an ELBO. The ELBO can be decomposed into one summand per part that describes the quality of the latent-to-observed mapping ("pose ELBO") and one summand for the dynamics mapping ("dynamics ELBO").

In our experiments, we set the number of body parts to $M = 2$ with one part corresponding to the upper body and one to the lower. By setting $M = 1$, we recover a variational version of the GPDM, denoted vGPDM.

3.2 Experiment

Our experiment was split in two parts, with the second part's parameter choices based on the results of the first part. Next, we describe the participants, the generation of stimuli, and then we detail the experimental paradigm.

3.2.1 Participants. We invited 31 participants to participate in the first part of the experiment via our participant management system (SONA System) and the university's mailing list. Due to technical problems, we excluded one participant from the analysis. The remaining 21 female and 9 male participants were between 19 and 44 years old ($\mu = 24.7a, \sigma = 5.8a$). Based on the results of this first part, we invited 26 participants to perform the second part of the experiment (19 female, age between 19 and 37 years, $\mu = 23.9a, \sigma = 4.2a$). All participants had normal or corrected-to-normal vision and received course credit or financial compensation (8€/h) for participation. The experimental procedures were approved by the local ethics committee and the study was conducted in accordance with the Declaration of Helsinki. Informed written consent was given by all participants prior to the experiment.

3.2.2 Stimuli. We employed a 10-camera PhaseSpace Impulse motion capture system to capture walking movements of an actor, and used our skeleton estimation software (Velychko and Endres 2017) to estimate a skeleton geometry with 18 joints, pose (Euler angles of each bone relative to the corresponding parental bone) and position and rotation of the pelvis bone. The results were stored in the Biovision Hierarchical Data format (bvh). From these data, we selected 49 sequences containing 3 gait cycles.

We used all 49 walking sequences to render the natural stimuli. Using the trained models, we generated 1,758 movement sequences (see next subsection), which served as artificial stimuli. Given the natural and generated bvh-files, we used Autodesk MotionBuilder to animate a gray avatar (see Figure 1) with body size and shape similar to the actor. We then rendered these animations into the videos used as stimuli. All resulting stimuli have

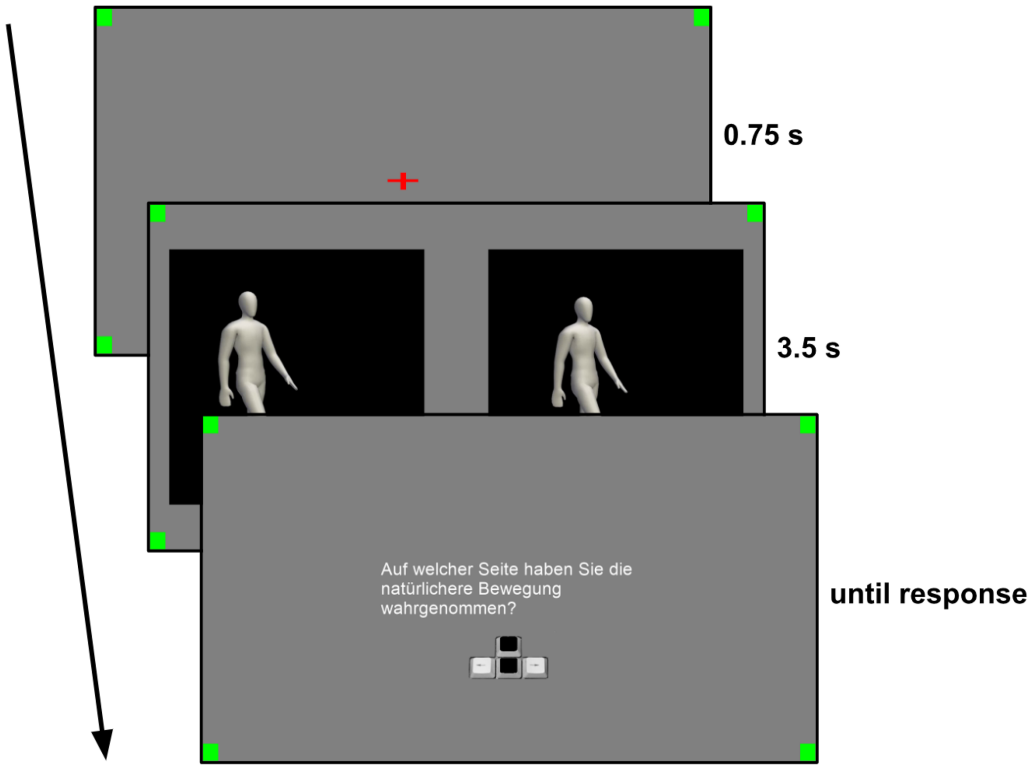


Fig. 1. Illustration of experimental procedure. Each trial begun with a fixation period of 0.75s. Then, participants watched simultaneous replays of natural and generated movements for 3.5s. After the presentation the participants were asked "On which side did you perceive the more natural movement?" and responded using the arrow keys of an keyboard.

a length of 3.5*s* with 60 frames per second. We supplied a demo video of some example trials in the supplementary material to give the reader a good impression of the stimuli and the task.

3.2.3 Stimulus Generation. We trained each MP model on nine gait sequences, and used the trained model to predict a tenth sequence. This enabled us to compute a leave-one-out cross-validation score for each model. Furthermore, the predicted sequence of joint angles was used for stimulus generation, as described above. Dynamical models were initialized with starting conditions taken from the training data. Sometimes the training procedure failed, because it is dependent on random initial values of the optimization algorithm. We hand-labeled obvious failures (e.g., sliding, limping, jerking, (see suppl. mat. first trial for an example)), excluding them from the data analysis, but retaining them to enable us to check the attention of the participants. Tables 1 and 2 summarize the tested models and ideal observer parameters. A more detailed description of the training procedure can be found in Velychko et al. (2018). We trained each model until the training target (ELBO or training MSE) did not change within machine precision anymore, but at most for one day. Most models were done training in a much shorter time.

3.2.4 Procedure. Participants were asked to distinguish between natural and generated movements in a two-alternative forced-choice task. For this, we designed an experiment using PsychoPy (Peirce 2009). During the experiment, participants were sitting in front of a 24-inch computer screen. After reading the written instructions, each trial proceeded as follows: (1) a fixation cross appeared for 0.75*s*, (2) followed by simultaneous side-by-side

presentation of generated and natural stimuli for 3.5*s*, and (3) finally collecting the participant's response, indicating on which side the more natural stimulus was perceived. Participants were instructed to use the arrow keys of a standard computer keyboard to submit their answer. They used the left index finger for the left arrow key, and the right index finger for the right arrow key. Both avatars were walking in the same direction, which was drawn randomly for each trial (see Figure 1).

Each participant of the first part of the experiment carried out 643 trials in four blocks, which took approximately 90 minutes. With these 643 trials, 119 models were evaluated: each participant rated 1 to 10 artificial stimuli randomly drawn from the total set of 10 artificial stimuli for each model. These were tested against a randomized repetition of 44 natural stimuli. To test whether participants simply memorized the natural stimuli during the experiment, we added 6 catch trials in the last quarter of the experiment where previously unused natural movements were tested against the known natural stimuli.

For the second part of the experiment, we split the total number of 629 trials into two conditions with 314 and 315 trials, allowing the participants to participate in one or both at their convenience. Participants were distributed equally among both conditions. Each condition was split into 7 blocks, with 30*s* pauses in between. After the first part of the experiment, we determined that memorization effects could be disregarded. Hence, we decided not to use catch trials in the second part. Sixty-seven models were tested in each condition. The available artificial stimuli for each model were distributed equally between conditions, and presented randomized for each participant.

3.3 Data Analysis

The rationale of the experiment is as follows: after simultaneous presentation of artificial and natural (motion-capture-based) human animations, the participant is forced to choose the one perceived as more natural. The answer is communicated via key press. In each trial i, we compute a random variable R_i from the key press, which assumes the value $r_i = 1$ if the participant was fooled by the artificially generated stimulus, and $r_i = 0$ otherwise. Thus, R_i is a Bernoulli distributed random variable. We assume the *confusion rate* p_i to be dependent on only the ideal observer parameters of the generated stimulus, such as number of basis-functions/MPs/IPs or model scores (see Section 3.1):

$$p(R_i = r_i) = p_i^{r_i}(1 - p_i)^{1-r_i} \tag{7}$$

We assume a conjugate p(oste)rior on the confusion rate p_i, i.e., a beta distribution, and compute error bars on p_i under this assumption. Please note that we decided to report the confusion rate as "success"-measure from the perspective of the model, which we want to evaluate, instead of reporting the discrimination ability of the participant $1 - p$ that is frequently used in the psychophysics literature.

Power Analysis. We would like to determine if the confusion rate of an artificial stimulus with a natural stimulus is less than chance. More precisely, denote hypothesis H0: $p_i \in [0.45, 0.55]$ and H1: $p_i \notin [0.45, 0.55]$. We choose the number of trials such that the falsehood of H0 is discovered with power 0.8 when H1 is true, i.e., $1 - P(\mathrm{H}_0|\mathrm{H}_1) = 0.8$. This yields a number of $N = 158$ trials for each parameter combination. Considering this number and our goal to test a wide range of parameter combinations (120 in total), the resulting number of trials is too large for a single participant. We therefore distribute the necessary trials across participants, excluding the possibility of inter-participant comparisons.

Logistic Regression. Each stimulus parameter combination is associated with scores S_i measuring the quality of the generated movement after training: the predictive mean squared error (MSE) for all models, ELBO for TMP, and v(c)GPDM models and dynamics- and pose-ELBO only for the v(c)GPDM models. We use logistic regression to find the relation between these model scores and the confusion rate:

$$p_i = \frac{c}{1 + \exp(w_0 + w_1 S_i),} \tag{8}$$

where $c \in [0, 0.5]$ reflects our assumption that the confusion rate can at best approach chance level. Assuming independence across N trials, we can compute the log-likelihood of all trials:

$$p(r_1, \ldots, r_N | w_0, w_1) = \log\left(\prod_{i=1}^{N} p(r_i)\right) \tag{9}$$

$$= \sum_{i=1}^{N} r_i \log(p_i) + \sum_{i=1}^{N} (1 - r_i) \log(1 - p_i). \tag{10}$$

We now learn the weights $(w_0^\star, w_1^\star)$ by maximizing the log-likelihood function using the `scipy.optimize.fmin_l_bfgs_b` routine (Jones et al. 2001). The gradients required for this optimizer are computed with `autograd` in `Python 3.6`.

Cross-Validation. We test the predictive capabilities of the different regressors S_i using n-fold cross-validation: the data set is split into n blocks, then weights are learned using n-1 blocks, and the log-likelihood of the left-out block is computed. This procedure is repeated n times, and the average left-out log-likelihood is used as score.

Logarithmic Likelihood-Ratio. We compare the predictive power of the different regressors against the null hypothesis: p_i is independent of S_i. We can now compute the cross-validatory log(likelihood-ratio) to evaluate the evidence for the statement "Model score S_i is more predictive of perceived naturalness than the best constant p_i".

4 RESULTS

We present the following results: participant evaluation, estimation of interesting parameter regimes, and finally comparison of model scores regarding their predictive power.

4.1 Evaluation of Participants

Attention Checks. During all parts of the experiment, we presented participants with attention check trials, where different, clearly unnatural stimuli had to be detected. We measured the detection rate of these stimuli. There were 17 attention check trials in the first part of the experiment and 15/14 in the second part's conditions. Over all trials, the detection rate was 98.0%. Three participants of the experiment had a detection rate of under 85%. These were excluded from further data analysis.

Catch Trials. During the first part of the experiment, we collected data from 162 catch-trials. 72 responses specified the previously unknown stimulus as more natural (44.4%). The probability that these responses are random, i.e. that they were generated by a Bernoulli process with $p = 0.5$ vs. $p \neq 0.5$ ($p \sim beta(1, 1)$) is ≈ 0.8. We are therefore fairly certain that the participants did *not* use memorization strategies for their response.

4.2 Estimating Regions of Interest in Parameter Space

We evaluated the perceived naturalness of 103 models using 976 stimuli during the first experiment (see Table 1). We collected 16902 trial responses from 27 participants in the first part of the experiment. Each participant completed 620 trials to estimate the confusion rate of models after exclusion of catch trials and attention checks. Across all trials, the confusion rate was 0.228. Please check the supplementary material to find a video with some example trials (with simulated random answers) to get an impression of the visual consequences for different models.

We used the results of this first part of the experiment to estimate more models of interest. For the TMP models, we decided after inspection of the confusion rate (Figure 2, left) to increase the number of MPs up to 15. Interestingly, the confusion rate seems to converge in the slightly hyper-realistic regime at $p \approx 0.55$. For the DMP models, we decided on testing numbers of basis function ranging from 50 to 100 (Figure 2, right). The confusion rate peaks at 80 basis functions. This does not coincide with the minimal predictive MSE, which is reached with 25 basis functions and increases from there on.

Table 1. Overview of Generated Trials for Each MP Model Type, Number of Attention Check Trials, and Number of Tested Parameter Combinations (After Excluding Attention Check Trials) in the First Part of the Experiment

MP model type	# Trials	# Att. checks	# Parameters combinations
vCGPDM	7,290	108	45
vGPDM	6,156	297	38
TMP	1,458	0	9
DMP	1,296	54	8
cGPDM (MAP)	270	0	1
GPDM (MAP)	270	0	1
Total	16,740	459	102

The confusion rates of the vGPDM models peak at (35, 10), (30, 20), (20, 20), (25, 35) (#IP Dynamics, #IPs pose) parameter combinations. These four parameter combinations are indistinguishable from natural stimuli (Figure 3, left). We estimated, by visual inspection, the location of the maximal confusion rate assuming that the confusion rate is described by a concave function of the parameters with additional noise. This yielded (25, 25) as the location of the global maximum.

The measured confusion rates of the vCGPDM models are equal at (20, 15) and (20, 20). We estimated (25, 20) to be a global maximum for the vCGPDM, in the same manner as for the vGPDM. Based on our power analysis and time budget, we decided on testing 67 parameter combinations for vGPDM and vCGPDM each. This way, we ended up testing 629 additional stimuli for the second part of the experiment (see Figure 4).

We also included GPDM and CGPDM models trained by MAP (maximum *a-posteriori*) instead of the ELBO. We measured confusion rates of 0.000 ± 0.004 for the MAP-GPDM, and 0.11 ± 0.02 for the MAP-CGPDM. These models were not tested again in the second part of the experiment. All resulting models are summarized in Table 2.

4.3 Predicting Perceived Naturalness

Using data from both experimental parts, we predicted the confusion rate from model scores via logistic regression. The results are shown in Figure 5 for TMP and DMP models and in Figure 6 for vGPDM and vCGPDM models. Depicted are the measured and predicted confusion rates for the tested models (columns), and different scores (rows). Furthermore, cross-validation results are summarized as log likelihood-ratio "ln K" of the prediction of the respective regressors versus the constant prediction (null-) hypothesis above each graph. Each "X" represents the confusion rate achieved by a unique parameter combination. The regression yields best results for the TMP models. MSE and ELBO of TMP models have similar predictive capabilities, as they are highly correlated in the investigated parameter regime. While the MSE also has predictive power for the v(C)GPDM models, the ELBO is not a suitable regressor. Inspection of the pose and dynamic terms of the ELBO reveals that this is due to the low score of the pose ELBO: $\ln K \approx -0.7$. The dynamic ELBO on the other hand even surpasses the MSE for the vCGPDM (Figure 6, left). Visual inspection of the logistic regression result for the DMP models shows that there is no simple sigmoidal relation between the perceptual validity and the DMPs MSE. This corresponds to the mismatch between MSE and confusion rate reported in Figure 2.

4.4 Comparing Best Models of Each MP-class

We plotted the confusion rate of all MP-models over the MSE in Figure 7. Even though a small MSE indicates better perceptual performance of the models, the relationship between MSE and confusion rate differs between the MP-model classes. For example, the vGPDM achieves high confusion rates even with high MSE.

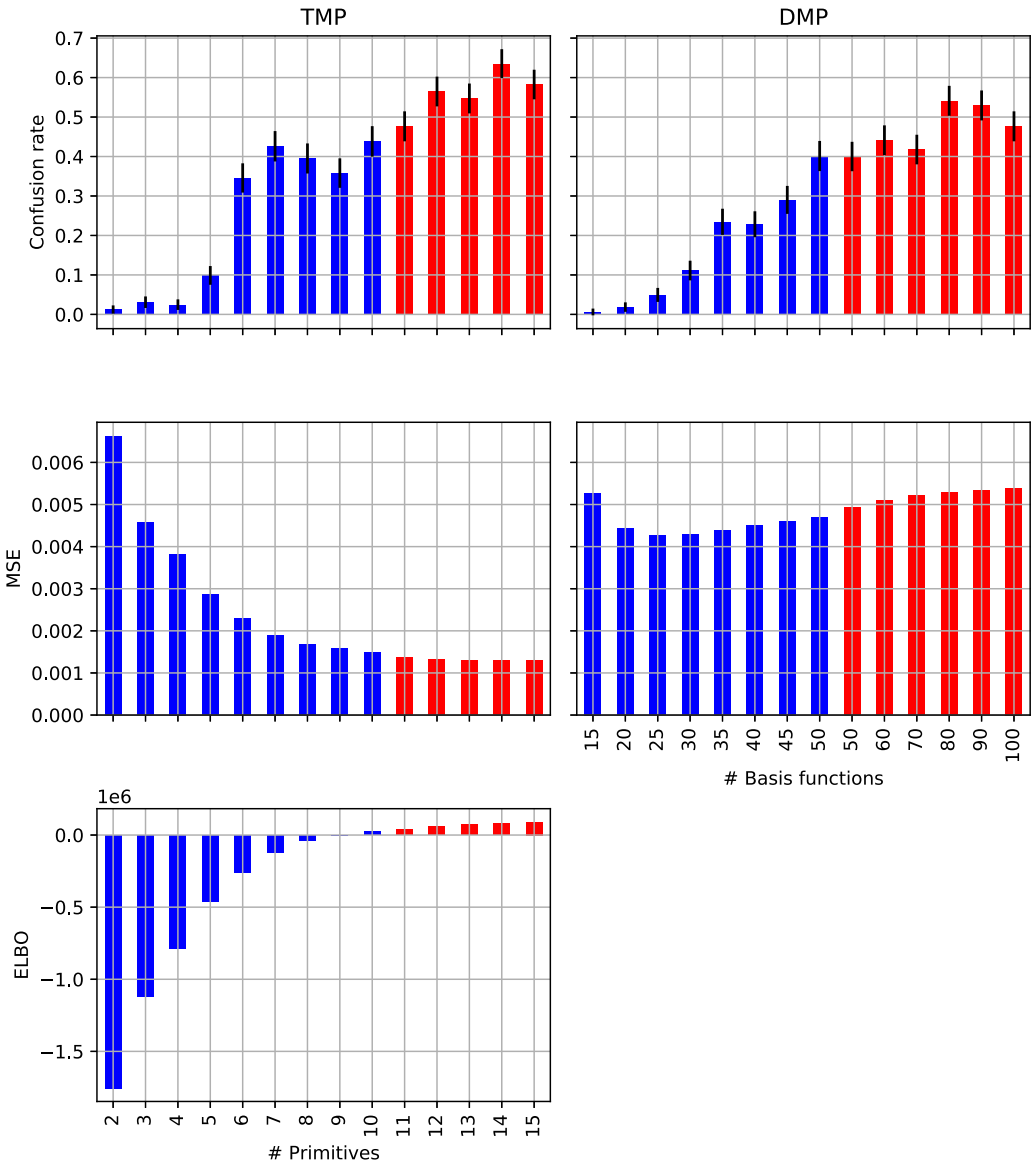


Fig. 2. Confusion rate, MSE, ELBO (from top to bottom) of TMP (left) and DMP (right) models for investigated model parameters. Data of first part of the experiment is colored blue, data of the second part is colored red.

For comparison of model performance we chose the best performing model of each MP-class, and computed the probabilities of all 6! = 720 many possible orderings of the models by confusion rate. We assumed beta(1,1) priors on the rate and a Bernoulli observation model, as before. The most probable ordering is TMP > vGPDM > DMP > vCGPDM > CGPDM(MAP) > GPDM(MAP) with a probability of 0.36. We computed marginal confusion rates and marginal pairwise ordering probabilities, see Figure 8. TMP, vGPDM, and DMP are comparable, while all other models are clearly worse. We used the same statistical model to test if the TMP's confusion rate is above 0.5, i.e., whether human participants perceive the model-generated stimulus as more natural than the natural one. Given our data, we are ≈ 0.99 sure of that.

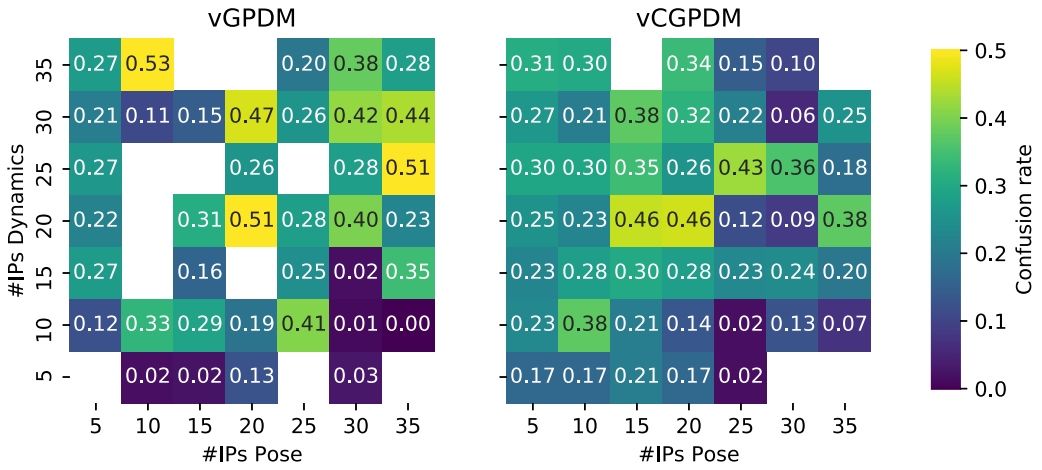


Fig. 3. Confusion rate of v(c)GPDM models in first part of experiment: Number of inducing points for the pose mapping on the x-axis, and for the dynamics mapping on the y-axis. The attention check parameter combinations are indicated by the white squares, where the model training procedure converged to obviously unnatural movements. Numbers on tiles are the measured confusion rates.

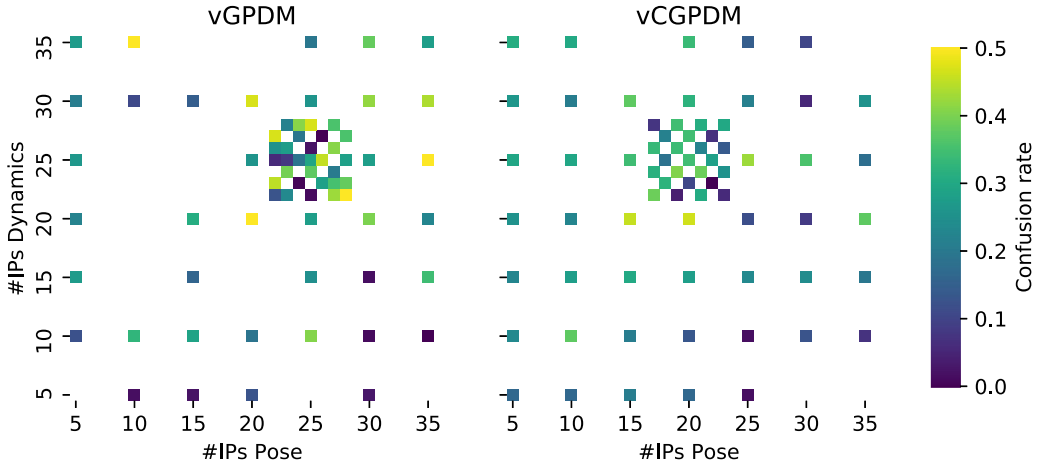


Fig. 4. Confusion rate of v(c)GPDM of first and second part of the experiment: Data of second part of the experiment are clustered around (25, 25) for vGPDM and (25, 20) for vCGPDM. Confusion rates are indicated by the same color-map as in Figure 3.

5 DISCUSSION

The tested MP models incorporate different (perceptual) predictive mechanisms: While TMPs determine the complete time course, the dynamical models make predictions for each next time-point from previous ones. The dynamical models therefore have advantages in feedback control applications where perturbations must be expected. TMPs, on the other hand, make perceptual predictions, as well as planning, easy, as there is no roll-out necessary to access the end-state of a movement.

The perceptually most valid, even hyper-realistic model is the variationally trained TMP. The shared representation between perception and production may therefore be more abstract: one dynamics model paired with

Table 2. Overview of Generated Trials for Each MP Model Type, Number of Attention Check Trials, and Number of Tested Parameter Combinations in the Second Part of the Experiment

MP model type	# Trials	# Att. checks	# Parameters combinations
vCGPDM	4,233	17	25
vGPDM	4,097	476	31
TMP	850	0	5
DMP	1,020	0	6
Total	10,200	493	67

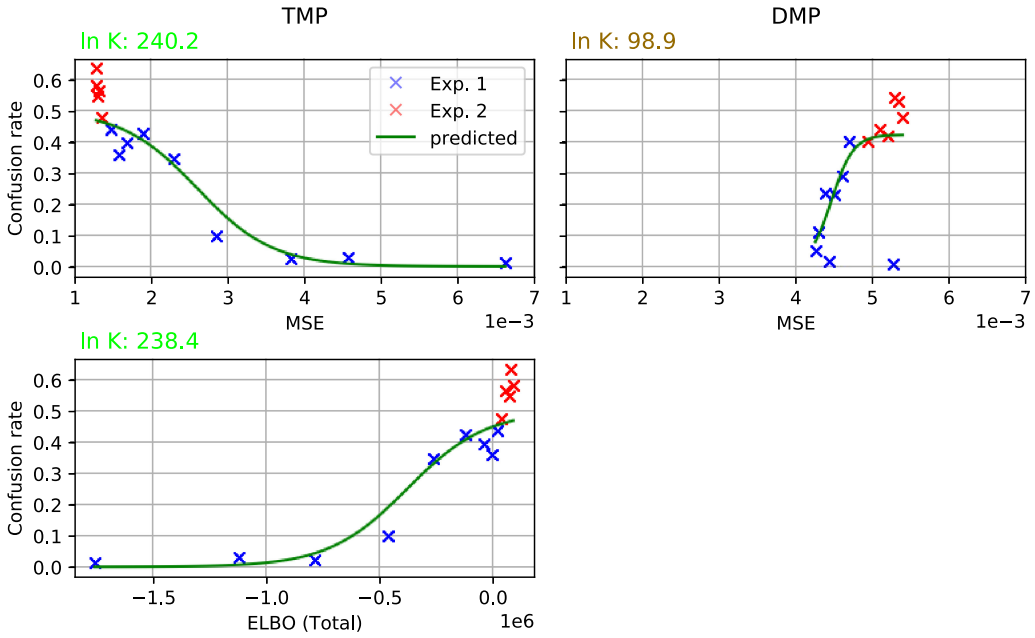


Fig. 5. Confusion rate of TMP (left) and DMP (right) models as function of model scores: MSE (top) and ELBO (bottom). Blue and red "X"s show confusion rates for model-parameters measured during experiment one and two. Green lines are predictions of the confusion rate (perceived naturalness) from the logistic regression using the regressor corresponding the abscissa label. Results of the cross-validation are summarized as log likelihood-ratio $\ln K$ in the top left corner of each plot, with the text color visualizing low (red) to strong (green) evidence in favour of the regressor being a good predictor of naturalness perception. See 3.3 for more detail.

a corresponding TMP model that encodes typical (unperturbed) solutions of the dynamics model, for fast perceptual predictions (Giese and Poggio 2000). Currently, we are preparing an experiment to compare TMP and dynamical MP models regarding their specific predictive mechanism employed in movement perception.

The vGPDM is still comparable to the TMP and the DMP, but that might change with more data. All other models are clearly worse. However, we are almost certain that the variationally approximated models are better than their MAP counterparts, which highlights the advantages of sparse variational posterior parametrizations.

We showed that approximate Bayesian model scores (ELBO, held-out MSE) can be used to predict the perceived naturalness of human animations. Assuming that humans are experts (i.e., nearly ideal observers) at perceiving their conspecifics' movements from noisy sensory input, it follows that their movement recognition performance

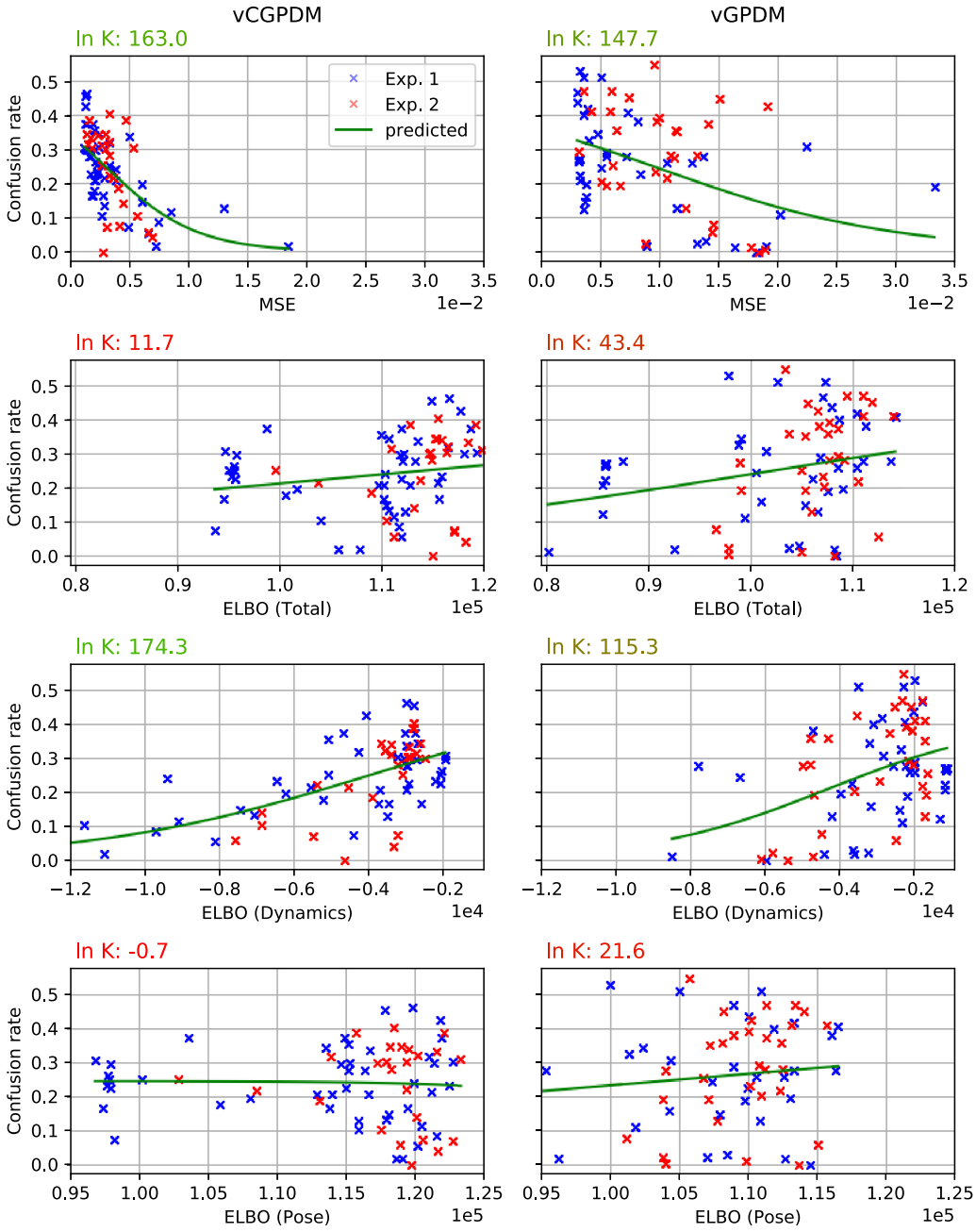


Fig. 6. Confusion rate of vCGPDM (left) and vGPDM (right) models as function of model scores: MSE, Total-, Dynamics-, Pose-ELBO (from top to bottom). Symbols have the same meaning as in Figure 5. See 3.3 for more detail.

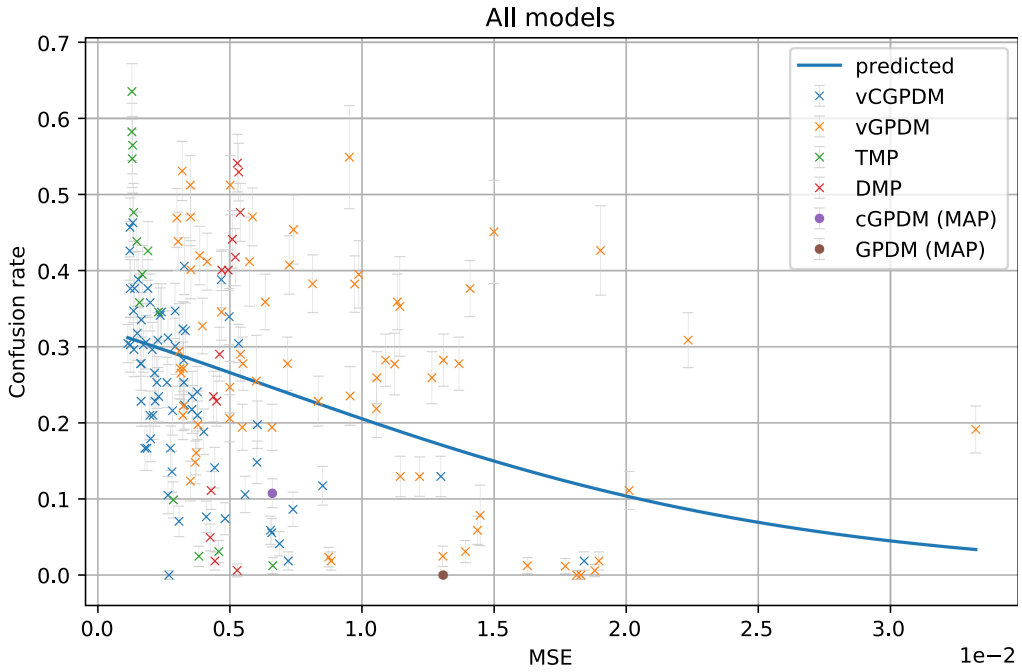


Fig. 7. Confusion rate of all models vs. test MSE and prediction learned over all models. Same data as in the first rows of Figures 5 and 6 plus cGPDM (MAP) and GPDM(MAP). Error bars denote beta standard deviation of the confusion rate.

should be near-Bayesian in general. Therefore, in particular, the perceived naturalness of a movement is expected to be predictable by approximate Bayesian model scores of the MPs. Our confirmation of this prediction adds evidence to the claim that human perception is nearly Bayes-optimal in many instances.

Comparison of total, dynamics, and pose ELBO as predictor for perceived naturalness of the v(C)GPDM models yields an interesting result: total ELBO is not a good predictor, because terms related to the latent-to-observed (pose) mapping apparently have no relevance for the perception of human animations. In contrast, dynamics ELBO scores indicate that a faithful dynamical mapping is more important than the pose mapping.

These computational level predictions might therefore also provide some insight into the perception of human animations on a algorithmic/mechanistic level: A feed-forward neural model (Giese and Poggio 2003) has been proposed arguing for the existence of separate motion and form pathways, where the motion pathway is performing a form of sequence recognition. Our results can thus be interpreted as additional evidence for importance of dynamics for perceiving human animations. Similar results have been derived from classical examinations of point light walkers (for a review, see Giese 2014): While local motion features form the simpler explanation for the perception of point light stimuli as biological motion than form features (Casile and Giese 2005), it has also been shown that biological motion perception can be induced in absence of local motion features (Beintema and Lappe 2002). For discrimination tasks, the information contained in the dynamics of the movement is more important than posture (Troje 2002).

Even though DMP models can generate highly realistic movement, a disadvantage is the unclear relation between MSE and perceptual validity. This finding demonstrates that the predictive MSE is not a sufficient indicator for perceptual performance: it is highly implausible that naturalness of a movement is evaluated by computing its point-wise deviation from an internal prototype for this movement.

The vGPDM performs comparable to the DMP, whereas the additional modular flexibility of the vCGPDM does not seem to be needed for our dataset: its best confusion rate is probably (86%) lower than that of the

	Parameter	Confusion rate	Error
TMP	14	0.64	±0.04
vGPDM	(22,28)	0.55	±0.07
DMP	80	0.54	±0.04
vCGPDM	(20,20)	0.46	±0.04
cGPDM (MAP)	-	0.11	±0.02
GPDM (MAP)	-	0.00	±0.00

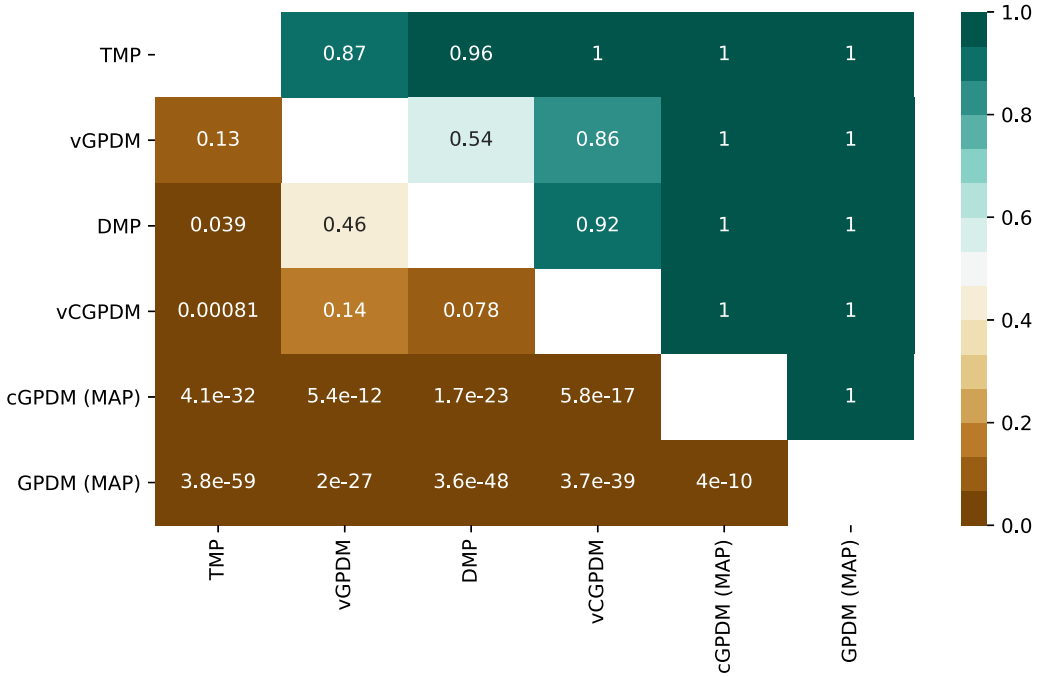


Fig. 8. Comparison of best models: (Top) Table of best models with corresponding parameter(-combinations), confusion rate and standard errors of beta posteriors. (Bottom) Bayesian ordering tests: probabilities that the best parameter combination of the models in the rows yields a higher confusion rate than the models in the columns. For example, the best TMP model (row) achieves a higher confusion rate than the best vGPDM model (column) with 87% certainty given our data.

vGPDM. This might also be due to the stochasticity in the training procedure: reachable optima depend on the random initial values of the optimization. Thus, the determined number of IPs where we suspected the perceptual optimum did not yield reliably high confusion rates or model scores for the second part of our experiment.

In our study, we only validated perceived naturalness of walking movements. We chose walking movements, because they are comparatively easy to model, yet highly important especially for animators. We are currently extending our investigation towards other, more complex movements, such as handling objects. Our hypothesis is that the main result–the Bayesian model score predicts naturalness perception–will generalize to these different movements as well, because at no point did we rely on features specific to walking.[2]

[2]The only exception is the specification of the DMP's attractor model, which is not important for our main results.

In our experimental paradigm we chose simultaneous side-by-side presentation of generated and natural movement videos. Simultanous presentation has two advantages: At any point in time there is a base-line for the participants. Presenting one after another would double the time of an already lengthy experiment. Still, the presentation time is short, thus the participants had to distribute their fixations across the two simultaneously presented videos. We will test and consider alternative paradigms, e.g., let participants rate naturalness on a scale. The gain of information per trial might be great enough to sacrifice the indistinguishability criterion. This might also enable inter-participant analysis, which is not possible in our paradigm, as described in 3.3 (Power Analysis).

6 CONCLUSIONS

Our study shows that MP models are capable of producing perceptually valid movements and we demonstrated that the prediction of naturalness is possible from model scores. These results add evidence for a shared MP-representation of action and perception and indicates the possibility of cheap, automated, and perceptually valid model selection for applications, e.g., in virtual reality. Finding a shared representation of MPs for perception and action could also provide a tool to study imitation learning in robots (Schaal 1999).

Congruent with previous studies, we found that parameters connected to dynamics are more relevant for perception than those connected with pose. This result could be useful to further improve generative models like the vCGPDM, and highlights the importance of prediction in the perception of human animations. While the Graphics Turing Test is a suitable tool for the estimation of perceived naturalness of movement, an analysis fixation data could shed some light on the features that drive this perception. Also, it would be interesting to determine what causes the hyper-realism of the TMP model.

Given that temporal and dynamical MP models have different advantages in movement planning and production, one of our current research directions is integrating such models into sensorimotor primitives, which are joint models of movement production and perception, with the aim of a computationally feasible instantiation of the common coding hypothesis. Sensory prediction during movement might not only be reflected in the movement itself, but also retrieved by an observer of biological movement, e.g., mime art. Applying such sensorimotor primitives to computer animation would enable a much more flexible interaction with avatars in virtual reality: Perceptually valid primitives could incorporate environmental constraints as well as the VR users movements, and be composed to form complex responsive behaviour of the avatar.

ACKNOWLEDGMENTS

We thank Olaf Haag for help with rendering of the stimuli and collecting data.

REFERENCES

Jaap Beintema and Markus Lappe. 2002. Perception of biological motion without local image motion. *Proceedings of the National Academy of Sciences* 99, 8 (April 2002), 5661–5663. DOI: https://doi.org/10.1073/pnas.082483699

Nikolai Bernstein. 1967. *The Co-ordination and Regulation of Movements.* Pergamon-Press. https://books.google.de/books?id=kX5OAQAAIAAJ

Bennett Bertenthal and Jeannine Pinto. 1994. Global processing of biological motions. *Psychological Science* 5, 4 (1994), 221–225. DOI: https://doi.org/10.1111/j.1467-9280.1994.tb00504.x

Antonino Casile and Martin A. Giese. 2005. Critical features for the recognition of biological motion. *Journal of Vision* 5, 4 (April 2005), 6–6. DOI: https://doi.org/10.1167/5.4.6

Enrico Chiovetto, Cristóbal Curio, Dominik Endres, and Martin A. Giese. 2018. Perceptual integration of kinematic components in the recognition of emotional facial expressions. *Journal of Vision* 18, 4 (April 2018), 13–13. DOI: https://doi.org/10.1167/18.4.13

Debora Clever, Monika Harant, Henning Koch, Katja Mombaur, and Dominik Endres. 2016. A novel approach for the generation of complex humanoid walking sequences based on a combination of optimal control and learning of movement primitives. *Robotics and Autonomous Systems* 83 (Sept. 2016), 287–298. DOI: https://doi.org/10.1016/j.robot.2016.06.001

Debora Clever, Monika Harant, Katja Mombaur, Maximilien Naveau, Olivier Stasse, and Dominik Endres. 2017. COCoMoPL: A novel approach for humanoid walking generation combining optimal control, movement primitives and learning and its transfer to the real robot HRP-2. *IEEE Robotics and Automation Letters* 2, 2 (2017), 977–984. DOI: https://doi.org/10.1109/LRA.2017.2657000

Andrea d'Avella, Philippe Saltiel, and Emilio Bizzi. 2003. Combinations of muscle synergies in the construction of a natural motor behavior. *Nature Neuroscience* 6, 3 (March 2003), 300–308. DOI: https://doi.org/10.1038/nn1010

Eran Dayan, Antonino Casile, Nava Levit-Binnun, Martin A. Giese, Talma Hendler, and Tamar Flash. 2007. Neural representations of kinematic laws of motion: Evidence for action-perception coupling. *Proceedings of the National Academy of Sciences* 104, 51 (Dec. 2007), 20582–20587. DOI: https://doi.org/10.1073/pnas.0710033104

Dominik Endres, Enrico Chiovetto, and Martin A. Giese. 2013. Model selection for the extraction of movement primitives. *Frontiers in Computational Neuroscience* 7 (2013), 185. DOI: https://doi.org/10.3389/fncom.2013.00185

Dominik Endres, Andrea Christensen, Lars Omlor, and Martin A. Giese. 2011. Emulating human observers with Bayesian binning: Segmentation of action streams. *ACM Transactions on Applied Perception (TAP)* 8, 3 (2011), 16:1–12.

Karl Friston. 2010. The free-energy principle: A unified brain theory? *Nature Reviews Neuroscience* 11, 2 (February 2010), 127–138. DOI: https://doi.org/10.1038/nrn2787

Martin A. Giese. 2014. Biological and body motion perception. *The Oxford Handbook of Perceptual Organization*. DOI: https://doi.org/10.1093/oxfordhb/9780199686858.013.008

Martin A. Giese and Tomaso Poggio. 2000. Morphable models for the analysis and synthesis of complex motion patterns. *International Journal of Computer Vision* 38 (June 2000), 59–73. DOI: https://doi.org/10.1023/A:1008118801668

Martin A. Giese and Tomaso Poggio. 2003. Neural mechanisms for the recognition of biological movements: Cognitive neuroscience. *Nature Reviews Neuroscience* 4, 3 (March 2003), 179–192. DOI: https://doi.org/10.1038/nrn1057

Simon Giszter. 2015. Motor primitives-New data and future questions. *Current Opinion in Neurobiology* 33 (Aug. 2015), 156–165. DOI: https://doi.org/10.1016/j.conb.2015.04.004

Simon Giszter, Emilio Bizzi, and Ferdinando A. Mussa-Ivaldi. 1992. Motor organization in the frog's spinal cord. In *Analysis and Modeling of Neural Systems*, Frank H. Eeckman (Ed.). Springer US, Boston, MA, 377–392. DOI: https://doi.org/10.1007/978-1-4615-4010-6_38

Jessica K. Hodgins, James F. O'Brien, and Jack Tumblin. 1998. Perception of human motion with different geometric models. 4, 4 (1998), 307–316. DOI: https://doi.org/10.1109/2945.765325

Bernhard Hommel, Jochen Müsseler, Gisa Aschersleben, and Wolfgang Prinz. 2001. The theory of event coding (TEC): A framework for perception and action planning. *Behavioral and Brain Sciences* 24 (2001), 849–937.

Auke Jan Ijspeert, Jun Nakanishi, Heiko Hoffmann, Peter Pastor, and Stefan Schaal. 2013. Dynamical movement primitives: Learning attractor models for motor behaviors. *Neural Computation* 25, 2 (Feb. 2013), 328–373. DOI: https://doi.org/10.1162/NECO_a_00393

Yuri P. Ivanenko, Richard E. Poppele, and Francesco Lacquaniti. 2004. Five basic muscle activation patterns account for muscle activity during human locomotion: Basic muscle activation patterns. *The Journal of Physiology* 556, 1 (April 2004), 267–282. DOI: https://doi.org/10.1113/jphysiol.2003.057174

Gunnar Johansson. 1994. Visual perception of biological motion and a model for its analysis. *Perceiving Events and Objects* 14 (1994), 185–207.

Eric Jones, Travis Oliphant, and Pearu Peterson. 2001. SciPy: Open source scientific tools for Python. [Online; accessed 2015-10-09].

David C. Knill and Alexandre Pouget. 2004. The Bayesian brain: The role of uncertainty in neural coding and computation. *Trends in Neuroscience* 27 (2004).

Michael D. McGuigan. 2006. Graphics turing test. *CoRR* abs/cs/0603132 (2006).

Lars Omlor and Martin A. Giese. 2011. Anechoic blind source separation using Wigner marginals. *Journal of Machine Learning Research* 12 (2011), 1111–1148.

Jonathan W. Peirce. 2009. Generating stimuli for neuroscience using PsychoPy. *Frontiers in Neuroinformatics* 2 (2009). DOI: https://doi.org/10.3389/neuro.11.010.2008

Felix Polyakov, Eran Stark, Rotem Drori, Moshe Abeles, and Tamar Flash. 2009. Parabolic movement primitives and cortical states: Merging optimality with geometric invariance. *Biological Cybernetics* 100, 2 (2009), 159.

Wolfgang Prinz. 1997. Perception and action planning. *European Journal of Cognitive Psychology* 9, 2 (June 1997), 129–154. DOI: https://doi.org/10.1080/713752551

Claire L. Roether, Lars Omlor, Andrea Christensen, and Martin A. Giese. 2009. Critical features for the perception of emotion from gait. *Journal of Vision* 9, 6 (June 2009), 15–15. DOI: https://doi.org/10.1167/9.6.15

Stefan Schaal. 1999. Is imitation learning the route to humanoid robots? *Trends in Cognitive Sciences* 3, 6 (June 1999), 233–242. DOI: https://doi.org/10.1016/S1364-6613(99)01327-3

Stefan Schaal. 2006. Dynamic movement primitives–a framework for motor control in humans and humanoid robotics. In *Adaptive Motion of Animals and Machines*, Hiroshi Kimura, Kazuo Tsuchiya, Akio Ishiguro, and Hartmut Witte (Eds.). Springer-Verlag, Tokyo, 261–280. DOI: https://doi.org/10.1007/4-431-31381-8_23

Krishna Shenoy, Maneesh Sahani, and Mark M. Churchland. 2013. Cortical control of arm movements: A dynamical systems perspective. 36, 1 (2013), 337–359. DOI: https://doi.org/10.1146/annurev-neuro-062111-150509

Yun Kyoung Shin, Robert W. Proctor, and E. John Capaldi. 2010. A review of contemporary ideomotor theory. *Psychological Bulletin* 136, 6 (Nov. 2010), 943–974. DOI: https://doi.org/10.1037/a0020541

David Sussillo, Mark M. Churchland, Matthew T. Kaufman, and Krishna V. Shenoy. 2015. A neural network that finds a naturalistic solution for the production of muscle activity. *Nature Neuroscience* 18, 7 (2015), 1025.

Nick Taubert, Andrea Christensen, Dominik Endres, and Martin A. Giese. 2012. Online simulation of emotional interactive behaviors with hierarchical gaussian process dynamical models. *Proceedings of the ACM Symposium on Applied Perception (ACM-SAP 2012)* (2012), 25–32. DOI: https://doi.org/10.1145/2338676.2338682

Emanuel Todorov and Michael I. Jordan. 2003. A minimal intervention principle for coordinated movement. In *Advances in Neural Information Processing Systems 15*, S. Becker, S. Thrun, and K. Obermayer (Eds.). MIT Press, 27–34. http://papers.nips.cc/paper/2195-a-minimal-intervention-principle-for-coordinated-movement.pdf.

Matthew Tresch, Philippe Saltiel, and Emilio Bizzi. 1999. The construction of movement by the spinal cord. *Nature Neuroscience* 2, 2 (Feb. 1999), 162–167. DOI: https://doi.org/10.1038/5721

Nikolaus Troje. 2013. What is biological motion? Definition, stimuli, and paradigms. *Social Perception: Detection and Interpretation of Animacy, Agency, and Intention.* 13–36. DOI: https://doi.org/10.7551/mitpress/9780262019279.003.0002

Nikolaus F. Troje. 2002. Decomposing biological motion: A framework for analysis and synthesis of human gait patterns. *Journal of Vision* 2, 5 (Sept. 2002), 2–2. DOI: https://doi.org/10.1167/2.5.2

Nikolaus F. Troje, Cord Westhoff, and Mikhail Lavrov. 2005. Person identification from biological motion: Effects of structural and kinematic cues. 67, 4 (2005), 667–675. DOI: https://doi.org/10.3758/BF03193523

Dmytro Velychko and Dominik Endres. 2017. A method and algorithm for estimation of pose and skeleton in motion recording systems with active markers (pending patent).

Dmytro Velychko, Dominik Endres, Nick Taubert, and Martin A. Giese. 2014. Coupling gaussian process dynamical models with product-of-experts kernels. In *Proceedings of the 24th International Conference on Artificial Neural Networks, Lecture Notes in Computer Science*, Vol. 8681. Springer, 603–610.

Dmytro Velychko, Benjamin Knopp, and Dominik Endres. 2018. Making the coupled Gaussian process dynamical model modular and scalable with variational approximations. *Entropy* 20, 10 (Sept. 2018), 724. DOI: https://doi.org/10.3390/e20100724

Jack Meng-Chieh Wang, David J. Fleet, and Aaron Hertzmann. 2008. Gaussian process dynamical models for human motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30, 2 (Feb. 2008), 283–298. DOI: https://doi.org/10.1109/TPAMI.2007.1167

Daniel M. Wolpert, Kenji Doya, and Mitsuo Kawato. 2003. A unifying computational framework for motor control and social interaction. *Philosophical Transactions of the Royal Society B: Biological Sciences* 358, 1431 (March 2003), 593–602. DOI: https://doi.org/10.1098/rstb.2002.1238

Received July 2019; accepted August 2019