Candidate Number: GBKC0

# Towards a BCH-Free Computation of the Composition of Stationary Velocity Fields

University College London
Medical Physiscs and Biomedical Engineering

A dissertation submitted in partial fulfillment
of the requirements for the degree of
**Master of Research**

August 12, 2015

Supervisor
**Tom Vercauteren**

Co-Supervisor
**Marc Modat**

Clinical Supervisor
**Jan Deprest**

# Abstract

In medical imaging registration, the study of transformations between patients images is underpinned by the study of geometrical transformations between anatomies. An important set of geometrical transformations that is utilized to model non-rigid deformations is provided by the set of diffeomorphisms (bijective differentiable maps with differentiable inverse).

Comparing two diffeomorphisms, as well as obtaining any meaningful statistics for these elements, is not a straightforward task.

The *log-Euclidean framework*, proposed to tackle this problem, uses some tools from differential geometry; it considers the set of diffeomorphisms with a Lie group structure, having a Lie algebra defined as the tangent space at the origin, as a vector space where to compute statistics.

In this vector space, the operation of composition of diffeomorphisms is not anymore available, but it is possible to define an operation, called *Lie log-composition*, that reflects the composition in the Lie group form the Lie algebra.

Aim of the research here presented is the study of numerical approximations for the computation of the Lie log-composition.


This document contains 15460 words (counted with detex).
Chapter $\mapsto$ Words : 1 $\mapsto$ 2304, 2 $\mapsto$ 3477, 3 $\mapsto$ 3795, 4 $\mapsto$ 1126, 5 $\mapsto$ 4025 6 $\mapsto$ 733

# Acknowledgments

It was not and it is still not easy for me the trail between the study of pure mathematics and the world of medical imaging and biomedical engineering. This route would have never been possible without guides, who have gone before me through the same tortuous paths and who accompanied me in the last year. The main contribution on this side arrived from Marco Lorenzi. With the commitment of a supervisor, he helped me significantly, not only in the development of this thesis, but especially in having introduced me to many of the problems a researcher have to deal with.

I am also grateful to the rowing fellows with whom I shared commitment, challenges and happiness in this first fascinating and demanding year at the GIFT-Surg Project: Francois Chadebecq, Pankaj Daga, Tom Doel, George Dwyer, Michael Ebner, Luis Herrera, Ioannis Kourouklides, Efthymios Maneas, Sacha Noimark, Rosalind Pratt, Marcel Tella, Gustavo Santos, Dzhoshkun Shakir, Guotai Wang and Maria A. Zuluaga. Fundamental was also the contribution of Jenny Nerny, Rebecca Holmes, Katie Konyn and Liz Zuzikova, that allow students and graduates to focus more on research than paper work.

For the help in the unknown land of infinite dimensional Lie algebra, I have a debt with professor Karl-H. Neeb and dott. Robert Gray.

A non academic, but not less important contribution came on different sides from Andrea Baglione, Gerardo Ballesio, Filippo Ferraris, Valeria Giacosa, Giuliano 'er Nuanda' De Rossi, Silvia Porter and Raoul Resta.

The eclectic buildings of UCL would have been unseen for me without Tom Vercauteren, Sebastien Ourselin and Gary Zhang. Their work and their decision, in a warm day of June 2014, to offer me their support - other than a desk, a laptop and a coffee machine - opened the greatest and most important opportunity I've ever had.

In classical music, it is well known that in every concert the two most important tunes are the first one and the last one. Following here the same rule I terminate acknowledging for the great love, effort and patience, Carole Sudre.

# Contents

# Chapter 1

# Introduction and Motivations

Medical image registration is a set of tools and techniques oriented to solve the problem of determining correspondences between two or more images acquired from patients scans. Its development is a creative field that has seen the application of a growing number of mathematical theories in the research of customizations, improvements of precision and reductions of computational time. It has a wide range of applications: for example it can be used in lungs motion correction [MHSK, MHM+11], in Alzheimer disease diagnoses [PCL+15, FF97, GWRNJ12], and image mosaicing [VPM+06, Sze94].

Determining correspondences between images is often presented as an ill-posed problem: transformations between anatomies are not unique, and the impossibility to recover spatial or temporal evolution of an anatomical transformation from few images over a long time, makes any validation a difficult, if not an impossible task.

The most important feature in image registration algorithms is the *deformation model*: the set of transformations chosen to model the anatomical deformations. Its choice is done in accordance with the task that the registration algorithm has to perform and with the nature of the objects represented by the images. See [ISNC03] for a presentation of the image registration framework and [SDP13] for a recent survey in medical image registration.

## 1.1 Choosing the Deformations: Diffeomorphisms

If the registration algorithm is meant to model physical transformations that preserve distances, orientations and angles, then the deformation model can be reduced to the group of rigid body transformations (see [Gal11] for a formal definition and applications to engineering). The consequent registration algorithm, called *rigid-registration algorithm*, will be suitable for example to compensate the motion in a rapid sequence of scans, or to investigate small differences that occurs in longitudinal studies.

If the algorithm is meant to model transformations that only preserves topology, then the transformations must allow more freedom than the one chosen for the rigid case. It is in this context that arises the idea of *non-rigid* registration. One of the possible registration models in this case is defined by the mathematical object of *diffeomorphism* over a compact subset $\Omega$ of $\mathbb{R}^d$ that represents the domain of the images. A diffeomorphism is defined as a bijective differentiable map from $\Omega$ to itself, with differentiable inverse, and is particularly well suited to model non-rigid deformation between images. Algorithms involving diffeomorphisms are called *diffeomorphic registration algorithms*.

Both rigid and diffeomorphic transformations belong to a *group structure* (see [Art11] for the abstract definition of group). In a group, only the operation of composition is available, and for mathematical reasons it is not possible to define any norm or meaningful mean of its elements.

One of the possible strategy to solve this problem is to consider the group with a structure of differentiable manifold compatible with the operation of composition - a *Lie group* - and to consider the linear approximation of its element in the tangent space - its *Lie algebra*; in this vector space it is possible to define a norm and therefore to compute statistics (see for example [Lee12, Arn06, War13, Car76, MTW73, HSSE09]).

The Pandora vase containing the idea of using the Lie algebra of the Lie group of diffeomorphisms for the computation of statistics in medical imaging was opened for the first time in 2006, with the name of *log-Euclidean framework* [AFPA06]. Authors proposed the numerical computation of the *Lie exponential* - the map that associate a vector in the vector space of the Lie algebra to the corresponding transformation in the Lie group - with a scaling and squaring algorithm. For its inverse, the *Lie logarithm* - the map that, when defined, associates a transformation in the Lie group to the corresponding vector in the Lie algebra - they proposed to use an inverse scaling and squaring algorithm.

These algorithms are not the only options available (see for example [BZO08], [BO08b]), but it is important to underline that once a transformation is mapped from the Lie group to the Lie algebra the operation of composition is not anymore available.

It is in this context that arises the abstract concept of *Lie log-composition*, whose numerical computations are the main aim of this research.

In the next section we will propose a formal definition of this operation, and in section 1.3 we will provide a short list of the possible applications of the log composition in medical imaging. The section 1.4 contains the outline of the thesis and the chapter ends with a brief remarks about the risks and issues involved when considering the infinite dimensional group of diffeomorphisms as a Lie group.

## 1.2 Introducing the Lie Log-composition and the BCH formula

Let exp be the Lie exponential and log the Lie logarithm, the *Lie log-composition* is defined for each couple of vectors $\mathbf{v}_1$, $\mathbf{v}_2$ belonging to the domain of exp as

$$\mathbf{v}_1 \oplus \mathbf{v}_2 := \log(\exp(\mathbf{v}_1) \circ \exp(\mathbf{v}_2)) \tag{1.1}$$

A schematic overview of this formula can be visualized in figure 1.1. The Lie group, indicated with $\mathbb{G}$, is represented by the gray surface, while the Lie algebra, indicated with $\mathfrak{g}$, is represented by the tangent plane at the identity $e$. Starting from the two vectors $\mathbf{v}_1$ and $\mathbf{v}_2$ in $\mathfrak{g}$, their Lie log-composition provides the vector that corresponds to the composition of the transformations corresponding to the initial vectors.

The analytic solution to the Lie log-composition is provided by the BCH formula (see [Hal15] for a formal introduction in the case of matrices). It was proved in 1947 by Dynkin [Dyn00] under the hypothesis that both the Lie exponential and Lie logarithm can be expressed in power series:

$$BCH(\mathbf{u}, \mathbf{v}) = \mathbf{u} + \mathbf{v} + \frac{1}{2}[\mathbf{u}, \mathbf{v}] + \frac{1}{12}([\mathbf{u}, [\mathbf{u}, \mathbf{v}]] + [\mathbf{v}, [\mathbf{v}, \mathbf{u}]]) - \frac{1}{24}[\mathbf{v}, [\mathbf{u}, [\mathbf{u}, \mathbf{v}]]] + ...$$

It consists in an infinite series of growing nested *Lie bracket* (a bilinear form defined within the Lie algebra structure, that reflects the geometrical curvature of the space of transformations - see in particular [MTW73] for a geometrical perspective on this definition).

There are many issues, both theoretical and practical that prevent from the use of the BCH formula for practical applications. The first and most obvious is that it is an infinite series whose truncations does not possess any asymptotic behaviour. In addition, even considering a large enough number of terms, the Lie bracket of two tangent vectors of the Lie group of diffeomorphisms involves the computation of the Jacobian matrices: this raises some numerical problems presented in section 5.2.3.

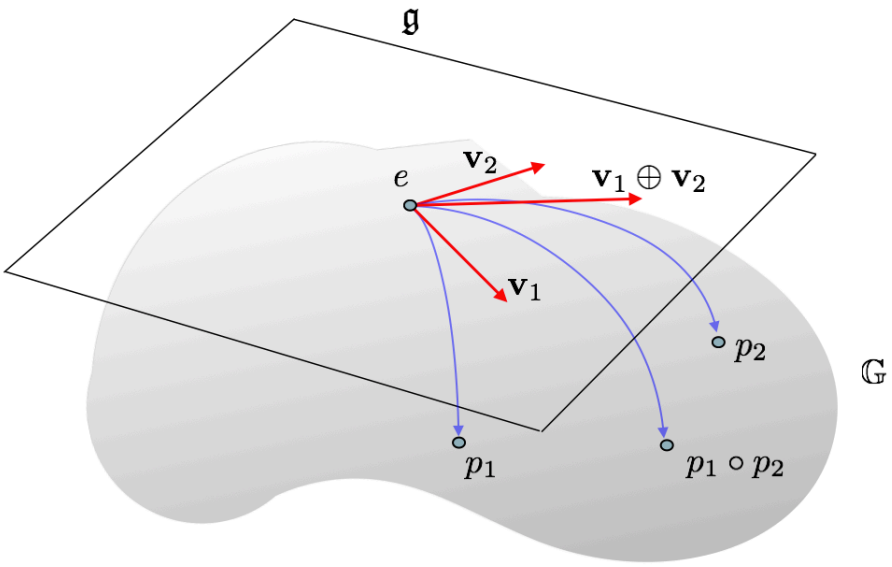



Figure 1.1: Graphical visualization of the Lie log-composition $\mathbf{v}_1 \oplus \mathbf{v}_2$. The gray surface represents a the Lie group of transformations between images and its tangent plane represents the Lie algebra, as the local linear approximation of the Lie group.

On the theoretical side, the proof of the BCH is based on the fact that Lie logarithm and exponential have to be expressed in power series. This happen only when the Lie group and its Lie algebra are subsets of a bigger algebra where scalar product, product and composition are compatible. This happens for matrix Lie groups (see [Hal15] for its definition), but in the case of diffeomorphisms, the matter is not free of deceptions (section 3.2 is devoted to the research of a bigger algebra that contains both Lie group of diffeomorphisms and its Lie algebra).

These are all good reasons to avoid the BCH formula when it is possible. In this thesis we will present two numerical methods for the computation of the Lie log-composition that does not rely on the truncated BCH - called here *BCH-free* methods. The first one, presented in section 2.4.2 is based on the Taylor expansion and its formulation holds in the case of matrices. The second one, presented in section 2.3.1 is based on a geometrical construction on the differentiable manifold of the transformations that exploit the concept of parallel transport.

## 1.3 Feasible Applications of the Log-composition in Medical Imaging

One of the reasons why mathematics is considered a powerful tool is consequence of the fact that a concept defined to solve a particular problem can, at the same time, be used to solve other problems from totally different origin and nature - this is known as the *unreasonable effectiveness of mathematics in the natural sciences* [Wig60]. In consequence of this, the more general and abstract is the tool developed, the more is versatile it become, but at the

same difficult to understand.

The Lie log-composition has been defined in this chapter from the problem of the computation of the statistics on the group of diffeomorphisms, but its use is not limited to perform only this task. In medical imaging there are several other situations in which its fast and accurate computation can be helpful:

1. Diffeomorphic demons [VPPA07] and log-demons algorithm [VPPA08]. In particular in the log-demons, the update at each of the iterative step is computed in the tangent space, using an equivalent formulation of the Lie log-composition.

2. Fast computation of the logarithm computation [BO08a]. Details of this algorithm are a part of this research and are discussed in chapter 4.

3. Calculus on diffusion tensor [AFPA06]. The logarithmic multiplication and the logarithmic scalar multiplication here defined, provide the Lie group with a structure of vector field. The Lie log-composition $\oplus$ here proposed provides the Lie algebra with a structure that reflects the composition of the group.

4. Image set classification [HWS$^+$]. As based on the log-euclidean framework, could exploit the property of having the group composition in the tangent space.

5. Computation of the the discrete ladder for the parallel transport [LP14a]. An equivalent of the Lie log-composition is utilized for the computation of the parallel transport.

## 1.4 Thesis' Outline

**Chapter 2** This chapter is devoted to the mathematical elements and tools involved in the numerical methods for the computation of the Lie log-composition.

**Chapter 3** In this chapter we introduce the two sets of transformations on which the numerical methods for the computation of the Lie log-composition will be tested and compared: the finite dimensional group of rigid body transformation and the infinite dimensional Lie group of diffeomorphisms.

**Chapter 4** The algorithm proposed in [BZO08] for computation of the Lie logarithm uses an equivalent formulation of the Lie log-composition as its main core. In this chapter we will see how the numerical methods developed in the previous chapter will be utilized in this context.

**Chapter 5** This chapter is devoted to the presentation of the numerical results applied to synthetic data and clinical images.

**Chapter 6** In the last chapter we summarize main results and we underline the further researches, both theoretical and practical, that can be done from this starting point.

Before moving to the next chapter, aimed to present the numerical techniques for its computation, it is worthed to spend a couple of words about the infinite dimensional Lie group of diffeomorphism.

## A Short Remark about the Lie Group of Diffeomorphisms

So far we have talked about the group of diffeomorphisms as a Lie group in a natural way, without underlining any particular feature of this structure. Actually, the topic of infinite dimensional Lie gorup is an open field of research whose development has not yet reached a definitive formalization. Aimed to present the theoretical problems and difficutlies as well

as how we deal with them in this Master Thesis, we retrace the main historical steps and some of the most significant approaches.

The first attempt to provide some handles to the group of diffeomorphisms for easy manipulation was done by Vladimir Arnold in 1966 [Arn66] (consider also the equivalent [Arn98], more readable for non-French speakers). To solve differential equation in hydrodynamic, the set of diffeomorphisms *Diff* is considered as a Lie group possessing a Lie algebra. This assumption is not formally investigated in accordance to the problem-oriented nature of this paper.

Subsequent steps in the exploration of the set of diffeomorphisms as a Lie group, and in the attempt of finding an abstract formalization can be found in [MA70, EM70, Omo70, Mic80, Les83]. A state of the art of infinite dimensional Lie group in the early eighties can be found in [Mil84a], while more recent results and applications on diffeomorphisms have been published in [OKC92, BHM10, Sch10, BBHM13].

Considering an infinite dimensional group as a differentiable manifold implies the idea of having each of its elements in local correspondence with some generalized "infinite-dimensional Euclidean" space. Attempts to set this correspondence showed that, the transition maps are smooth over the Banach spaces, results that led to the idea of Banach Manifolds. It has been shown that the group of diffeomorphisms defined as a manifold does not belongs to the category of Banach manifold but requires an even more general space on which the transition maps are smooth: the *Frechet space* (see [KW08]). Here, important theorems from analysis, as the inverse function theorem, the Frobenius theorem, or the main results from the Lie group theory in a finite dimensional settings, as Lie correspondence theorems, do not hold anymore.

These difficulties led some researchers to approach the set of diffeomorphisms from other perspectives: for example, instead of treating *Diff* as a group equipped with differential structures, it is seen as a quotient of other well behaved groups [Woj94]. In other cases, as in [MA70] first and in [Mil84b] later, Banach spaces are substituted with more general locally convex spaces.

For the medical imaging purposes, it is not necessarily to consider the general theory of infinite dimensional manifolds; we can take into account only diffeomorphisms that are interesting for our practical applications, i.e. the one defined on a compact subset $\Omega$ of $\mathbb{R}^d$. Moreover, without denying the importance of fundamentals and underestimating the doors that research in infinite dimensional Lie group theory may open, we will approach diffeomorphisms in as similar way of what has been done in set theory: we will use a *naive approach* to infinite dimensional Lie group. Here the fundamental definition of infinite dimensional Lie group is a generalization of the finite dimensional case of matrices, and it is left more to the intuition than to a robust formalization.

# Chapter 2

# Tools from Differential Geometry

## 2.1 A Lie Group Structure for the Set of Transformations

We consider every group $\mathbb{G}$ as a group of transformations acting on $\mathbb{R}^d$, having in mind the particular case $d = 2, 3$ for 2-dimensional or 3-dimensional images. We will focus our attention to transformations defined by matrices or diffeomorphism. Not only do they have the structure of a group, but they have as well the structure of a Lie group. They posses maximal atlas that makes them differentiable manifold, in which the composition of two transformations and the inversion of each transformation are well defined differentiable maps

$$\begin{aligned} \mathbb{G} \times \mathbb{G} &\longrightarrow \mathbb{G} \\ (x, y) &\longmapsto xy^{-1} \end{aligned}$$

Generally speaking, differential geometry is a technique to use the well known calculus features and operators on spaces different from the usual $\mathbb{R}^n$. Adding the differentiable structure to a group of transformations provides new handles to hold and manipulate it; in particular it provides the opportunity to define a tangent space to each point of the group and consequently a *fiber bundle* (the disjoint union of all of the tangent space), a space of vector fields, a set of flows and one parameter subgroup.

We will refer to [Car76] and [Lee12] for the definitions and concepts of differential geometry and [Car92] for definition and concepts of Riemannian geometry.

## 2.2 Lie Exponential, Lie logarithm, Lie log-composition and the BCH formula

Let $\mathbf{v}$ be an element in the tangent space $\mathfrak{g}$ of the (compact) Lie group $\mathbb{G}$. The *Lie exponential* is defined as

$$\begin{aligned} \exp : \mathfrak{g} &\longrightarrow \mathbb{G} \\ \mathbf{v} &\longmapsto \exp(\mathbf{v}) = \gamma(1) \end{aligned}$$

where $\gamma : [0, 1] \to \mathbb{G}$ is the unique one-parameter subgroup of $\mathbb{G}$ having $\mathbf{v}$ as its tangent vector at the identity. The identity is indicated with $e$ for the general case, $I$ for matrices and 1 for diffeomorphisms. The exponential map satisfies the following properties:

1. $\exp(t\mathbf{v}) = \gamma(t)$.
2. $\exp(\mathbf{v}) = e$ if $\mathbf{v} = \mathbf{0}$.
3. $\exp(\mathbf{v}) \circ \exp(-\mathbf{v}) = e$
4. As a direct consequence of the definition here provided, based on the one parameter subgroup, it follows that:

$$\exp((t+s)\mathbf{v}) = \gamma(t+s) = \gamma(t) \circ \gamma(s) = \exp(t\mathbf{v}) \exp(s\mathbf{v})$$

   This is in particular one of the reasons that justifies the name exponential for a maps between structures in differential geometry.

5. $\exp(\mathbf{v})$ is invertible and $(\exp(\mathbf{v}))^{-1} = \exp(-\mathbf{v})$.
6. $\exp(\mathbf{u}+\mathbf{v}) = \lim_{m\to\infty}(\exp(\frac{\mathbf{u}}{m}) \circ \exp(\frac{\mathbf{v}}{m}))^m$
7. exp is a local isomorphism: it is an isomorphisms between a neighborhood of $\mathbf{0}$ in $\mathfrak{g}$ to a neighborhood of $e$ in $\mathbb{G}$.
8. If $\exp(\mathbf{w}) = \exp(\mathbf{u}) \exp(\mathbf{v})$ then

$$\exp(-\mathbf{w}) = \exp(-\mathbf{v}) \exp(-\mathbf{u}) \tag{2.1}$$

*Proof.* We will prove only the last statement leaving the others to the literature. The hypothesis $\exp(\mathbf{w}) = \exp(\mathbf{v}) \circ \exp(\mathbf{u})$ follows the subsequent chain of implications (each algebraic passage involves a geometrical construction, not shown here for brevity):

$$\begin{aligned}
\exp(\mathbf{w}) &= \exp(\mathbf{v}) \circ \exp(\mathbf{u}) \\
\exp(-\mathbf{w}) \circ \exp(\mathbf{w}) &= \exp(-\mathbf{w}) \circ \exp(\mathbf{v}) \circ \exp(\mathbf{u}) \\
e &= \exp(-\mathbf{w}) \circ \exp(\mathbf{v}) \circ \exp(\mathbf{u}) \\
\exp(-\mathbf{u}) &= \exp(-\mathbf{w}) \circ \exp(\mathbf{v}) \\
\exp(-\mathbf{u}) \circ \exp(-\mathbf{v}) &= \exp(-\mathbf{w})
\end{aligned}$$

□

When we deal with a matrix Lie group of dimension $n$, the composition in the Lie group consists in the matrix product and we have the following remarkable properties [Hal15], [Kir08]:

1. for all $\mathbf{v}$ in a matrix Lie algebra $\mathfrak{g}$:

$$\exp(\mathbf{v}) = \sum_{k=0}^{\infty} \frac{\mathbf{v}^k}{k!} \tag{2.2}$$

2. If $\mathbf{u}$ and $\mathbf{v}$ are commutative then $\exp(\mathbf{u}+\mathbf{v}) = \exp(\mathbf{u}) \exp(\mathbf{v})$.
3. If $\mathbf{c}$ is an invertible matrix then $\exp(\mathbf{c}\mathbf{v}\mathbf{c}^{-1}) = \mathbf{c} \exp(\mathbf{v})\mathbf{c}^{-1}$.
4. $\det(\exp(\mathbf{v})) = \exp(\text{trace}(\mathbf{v}))$
5. For any norm, $||\exp(\mathbf{v})|| \leq \exp(||\mathbf{v}||)$.

The idea of defining an inverse of the Lie exponential leads to the idea of the Lie logarithm, defined as

$$\begin{aligned} \log : \mathbb{G} &\longrightarrow \mathfrak{g} \\ \varphi &\longmapsto \log(\varphi) = \mathbf{v} \end{aligned}$$

where $\mathbf{v}$ is the tangent vector having $\varphi$ as it exp.
If $\mathbb{G}$ is a matrix Lie group of dimension $n$, the following properties hold:

1. for all $\varphi$ in the matrix Lie group $\mathbb{G}$:

$$\log(\varphi) = \sum_{k=1}^{\infty} (-1)^{k+1} \frac{(\varphi - I)^k}{k} \tag{2.3}$$

2. For any norm, and for any $n \times n$ matrix $\mathbf{c}$, exists an $\alpha$ such that

$$||\log(I + \mathbf{c}) - \mathbf{c}|| \leq \alpha ||\mathbf{c}||^2 \tag{2.4}$$

3. For any $n \times n$ matrix $\mathbf{c}$ and for any sequence of matrix $\{\mathbf{d}_j\}$ such that $||\mathbf{d}_j|| \leq \alpha / j^2$ it follows:

$$\lim_{k \to \infty} \left(I + \frac{\mathbf{c}}{k} + \mathbf{d}_k\right)^k = \exp(\mathbf{c}) \tag{2.5}$$

The *Lie log-composition* (the name Lie is added because based on the Lie logarithm and Lie exponential maps) is defined here as the inner binary operation on the Lie algebra that reflects the composition on the lie group:

$$\begin{aligned} \oplus : \mathfrak{g} \times \mathfrak{g} &\longrightarrow \mathfrak{g} \\ (\mathbf{v}_1, \mathbf{v}_2) &\longmapsto \mathbf{v}_1 \oplus \mathbf{v}_2 = \log(\exp(\mathbf{v}_1) \circ \exp(\mathbf{v}_2)) \end{aligned}$$

The following properties holds for the Lie log-composition:

1. $\mathfrak{g}$ with the Lie log-composition $\oplus$ is a local topological non-commutative group (local group for short):
   (a) $(\mathbf{u}_1 \oplus \mathbf{u}_2) \oplus \mathbf{u}_3 = \mathbf{u}_1 \oplus (\mathbf{u}_2 \oplus \mathbf{u}_3)$ for all $\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3$ in $\mathfrak{g}$.
   (b) $\mathbf{u} \oplus \mathbf{0} = \mathbf{0} \oplus \mathbf{u} = \mathbf{u}$ for all $\mathbf{u}$ in $\mathfrak{g}$.
   (c) $\mathbf{u} \oplus (-\mathbf{u}) = \mathbf{0}$ for all $\mathbf{u}$ in $\mathfrak{g}$.
2. For all $t, s$ real:

$$(t\mathbf{u}) \oplus (s\mathbf{u}) = (t + s)\mathbf{u}$$

   And in particular, if the Lie algebra $\mathfrak{g}$ has dimension 1 the local group structure is compatible with the additive group of the vector space $\mathfrak{g}$.

The Backer-Campbell-Hausdorff formula, or BCH, provides the analytic solution to the Lie log-composition:

$$BCH(\mathbf{u}, \mathbf{v}) = \mathbf{u} + \mathbf{v} + \frac{1}{2}[\mathbf{u}, \mathbf{v}] + \frac{1}{12}([\mathbf{u}, [\mathbf{u}, \mathbf{v}]] + [\mathbf{v}, [\mathbf{v}, \mathbf{u}]]) - \frac{1}{24}[\mathbf{v}, [\mathbf{u}, [\mathbf{u}, \mathbf{v}]]] + ...$$

The name of the formula is after the mathematicians H. F. Baker, J. E. Campbell, and F. Hausdorff, and ironically does not refer to Dynkin, who originally developed the proof of the previous equality in 1947 [Dyn00]. This formula has already been used in medical imaging, for example in [LP14b], [VPPA08] and [BO08b]. In these papers the computation has been performed considering only the first two or three terms. As we will see in section 3.2.5 and 5.2.3 this approximation can be problematic.

## 2.3 Affine Exponential, Affine Logarithm and Parallel Transport: Definitions and Properties

Considering a Lie Group $\mathbb{G}$ with a connection $\nabla$, the vector field $\nabla_U(V)$ associates at each point of the manifold the projection on the tangent plane of the derivative of $U$ in the direction of $V$.

One of the interesting consequences of the definition of the connection is that it allows to define *geodesics* and curvature on the manifold without relying on any Riemannian metric. If a Riemannian metric is also defined on the manifold $\mathbb{G}$, then geodesics defined by the metric coincides with the geodesics defined by the connection only for the particular case of Levi-Civita connection (see [Car92]).

A curve $\gamma : [0,1] \to \mathbb{G}$ such that $\gamma(0) = p$ and $\gamma(1) = q$ is a *geodesic* defined by the connection $\nabla$ if

$$\nabla_{\dot\gamma}\dot\gamma = 0 \tag{2.6}$$

From this definition arises a new kind of exponential from the Lie algebra to the Lie group: given the point $p$ and the tangent vector at this point $\mathbf{v} \in T_p\mathbb{G} \simeq \mathfrak{g}$ we define

$$\begin{aligned} \exp : \mathbb{G} \times \mathfrak{g} &\longrightarrow \mathbb{G} \\ (p, \mathbf{v}) &\longmapsto \exp_p(\mathbf{v}) = \gamma(1; p, \mathbf{v}) \end{aligned}$$

such that the curve $\gamma(t; p, \mathbf{v}) = \gamma(t)$ on $\mathbb{G}$ is the unique geodesic that satisfies $\gamma(0) = p$ and $\dot\gamma(0) = \mathbf{v}$. It is called *affine exponential* and differs from the Lie exponential map previously introduced by the fact that it is applied at the generic point $p$ of the Lie group and not only at the identity $e$.
The inverse of the affine exponential, the *affine logarithm* is defined as:

$$\begin{aligned} \log : \mathbb{G} \times \mathbb{G} &\longrightarrow T_p\mathbb{G} \simeq \mathfrak{g} \\ (p, q) &\longmapsto \log_p(q) = \mathbf{v} \end{aligned}$$

where $\mathbf{v}$ is the tangent vector at the point $p$ of the geodesic $\gamma$ on $\mathbb{G}$ that satisfies $\gamma(0) = p$ and $\gamma(1) = q$. Interestingly, if $\nabla$ is a Cartan connection the Lie exponential and the Lie logarithm coincide with the affine exponential and the affine logarithm at the identity.

For further details and properties we refer to the literature; in this introduction we wish to provide only the intuitive idea that it is possible to move on the fiber bundle of the Lie group, transporting in some sense a tangent vector defined at the identity on another tangent space. Certainly the Lie group possesses a unique Lie algebra, as the tangent space at some point (the group's identity by convention), but two different tangent spaces (so two times the same isomorphic Lie algebra structure) may not have the basis oriented in the same direction.

### 2.3.1 An introduction to Parallel Transport: Surfing on the Fiber Bundle

In this section we introduce the concepts of vector field and parallel transport for the Lie group $\mathbb{G}$. For an introduction to parallel transport in the general case we refer to [MTW73], [Kne51], [KMN00]; for medical imaging applications [LAP11], [PL$^+$11], [LP13] and [LP14b]. On these definitions, again borrowed from differential geometry, relies a method for the computation of the Lie log-composition developed in this research for the first time.

**Definition 2.3.1.** Let $\mathbb{G}$ be a finite dimensional connected Lie group, a *vector field* over $\mathbb{G}$ is a function that assigns at each point $p$ of $\mathbb{G}$, a tangent vector $V_p$ in the tangent space $T_p\mathbb{G}$, such that $V_p$ is differentiable respect to $p$. If $(C; x_1, \ldots, x_n) = (C, \psi)$ is a local chart of $\mathbb{G}$, neighborhood of $p$, then $V_p$ can be expressed locally as $V_p = \sum_{i=1}^n v_i(p) \frac{\partial}{\partial x_i}\Big|_p$ for $v_i \in \mathcal{C}^\infty(C)$.

**Definition 2.3.2.** Let $\mathbb{G}$ be a Lie group defined with a connection $\nabla$ and $V$ a $\mathcal{C}^\infty$ vector field defined over $\mathbb{G}$. Given $p, q \in \mathbb{G}$ and $\gamma : [0,1] \to \mathbb{G}$ such that $\gamma(0) = p$ and $\gamma(1) = q$, the vector $V_p \in T_p\mathbb{G}$, is *parallel transported along* $\gamma$ up to $T_q\mathbb{G}$ if $V$ satisfies

$$\nabla_{\dot\gamma} V_{\gamma(t)} = 0 \qquad \forall t \in [0,1]$$

The *parallel transport* is the function that maps $V_p$ from $T_p\mathbb{G}$ to $T_q\mathbb{G}$ along $\gamma$:

$$\begin{aligned} \Pi(\gamma)_p^q : T_p\mathbb{G} &\longrightarrow T_q\mathbb{G} \\ V_p &\longmapsto \Pi(\gamma)_p^q(V_p) = V_q \end{aligned}$$

Consequence of this definition is that a vector belonging to the tangent space at the identity can be transported on a different tangent space of the manifold, maintaining its direction from the old to the new coordinate reference respect to a chosen curve. Each element of the fiber bundle, that can be reached by a curve from the origin, becomes reachable also by any tangent vector at the identity.

A way of moving vectors from an arbitrary tangent space to the tangent space at the identity is expressed in the *change of base formulas* for affine exponential and logarithm [APA06]:

$$\log_p(q) = DL_p(e)\log_e(q) \tag{2.7}$$

$$\exp_p(\mathbf{u}) = p \circ \exp_e(DL_{p^{-1}}(e)\mathbf{u}) \tag{2.8}$$

The left-translation $L_p$ provides a canonical curves for transporting vectors, expressed as the integral curve of the tangent vector field on the manifold of transformations defined by the push forward of $L_p$, indicated here with $DL_p$.

We will now explore how the parallel transport and the affine exponential behave when expressed as a composition and when there is a change of signs.

**Property 2.3.1** (Inversion)**.** $\mathbb{G}$ Lie group, $\nabla$ connection, $p, q \in \mathbb{G}$. Given $\gamma$ such that $\gamma(0) = p$, $\gamma(1) = q$ and $\mathbf{u} \in T_p\mathbb{G}$, we have:

1. $\Pi(\gamma)_p^q(-\mathbf{u}) = -\Pi(\gamma)_p^q(\mathbf{u})$
2. $q = \exp_p(\mathbf{u}) \iff p = \exp_q(-\Pi(\gamma)_p^q(\mathbf{u}))$

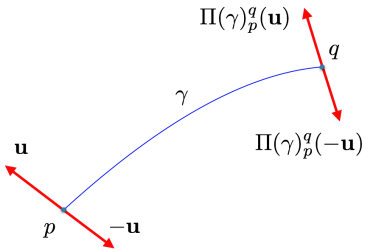



Figure 2.1: First inversion property.

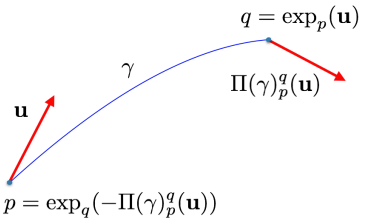



Figure 2.2: Second inversion property.

*Proof.* The first statement is a consequence of the fact that parallel transport is reversible, conserve the parallelism and it is invariant respect to a norm:

i) $\Pi(-\gamma)_q^p\Big(\Pi(\gamma)_p^q(\mathbf{u})\Big) = \mathbf{u}$ where $-\gamma$ corresponds to $\gamma$ walked in the opposite direction.

ii) $\Pi(\gamma)_p^q(\mathbf{u})$ is parallel in the same tangent space to $\Pi(\gamma)_p^q(\lambda\mathbf{u})$ for any nonzero $\lambda$.

iii) $||\Pi(\gamma)_p^q(\mathbf{u})|| = ||\mathbf{u}||$

To prove the second statement, if $q = \exp_p(\mathbf{u})$ then it exists a curve $\gamma$ that is a geodesics and connects $p$ with $q$:

$$\exp_p(\mathbf{u}) = \gamma(1; \mathbf{u}, p) \qquad \nabla_{\dot\gamma}\dot\gamma = 0 \qquad \gamma(0) = p \qquad \gamma(1) = q$$

On the other side, if $p = \exp_q(-\Pi(\gamma)_p^q(\mathbf{u}))$, then it exists a curve $\beta$ that is a geodesic and connect $q$ with $p$:

$$\exp_q(-\Pi(\gamma)_p^q(\mathbf{u})) = \beta(1; -\Pi(\gamma)_p^q(\mathbf{u}), q) \qquad \nabla_{\dot\beta}\dot\beta = 0 \qquad \beta(0) = q \qquad \beta(1) = p$$

Since there is a unique curve that satisfies the condition of being geodesic between two points, we have $\gamma = -\beta$. Therefore, if $q = \exp_p(\mathbf{u})$, then

$$p = \gamma(0; \mathbf{u}, p) = \beta(1; -\Pi(\gamma)_p^q(\mathbf{u}), q)$$

which implies $p = \exp_q(-\Pi(\gamma)_p^q(\mathbf{u}))$. On the other side, if $p = \exp_q(-\Pi(\gamma)_p^q(\mathbf{u}))$, then

$$q = \beta(0; -\Pi(\gamma)_p^q(\mathbf{u}), q) = \gamma(1; \mathbf{u}, p)$$

which implies $q = \exp_p(\mathbf{u})$. □

**Property 2.3.2.** Let $\mathbb{G}$ be a finite dimensional connected Lie group defined with a Cartan connection $\nabla$ and $\mathbf{u}$ tangent vector in $T_e\mathbb{G}$. Let $\gamma$ be a geodesic defined on $\mathbb{G}$ such that $\gamma(0) = e$, $\dot\gamma(0) = \mathbf{u}$ and $p = \gamma(1)$. Let $\beta$ be the curve over $\mathbb{G}$ defined as $\beta(t) = p \circ \gamma(t)$, then the two following conditions hold:

1. If $\nabla$ is a Cartan connection then $\beta$ is a geodesic.
2. For $\mathbf{u}_p := DL_p(e)(\mathbf{u}) \in T_p\mathbb{G}$, push forward of the left-translation:

$$\exp_p(t\mathbf{u}_p) = p \circ \exp_e(tDL_{p^{-1}}(p)(\mathbf{u}_p)) = p \circ \exp_e(t\mathbf{u}) \tag{2.9}$$

*Proof.* The first statement belongs to the general theory and it is not proved here: geodesics are left-invariant for a Cartan connection (see [Car92]). To prove the second statement we consider the properties of $\beta$ that directly follows from the definition:

$$\beta(0) = p \circ e = p$$
$$\dot\beta(0) = DL_p(e)\mathbf{u} \in T_p\mathbb{G}$$

For simplicity $\dot\beta(0)$ was indicated with $\mathbf{u}_p$. Considering $\beta(1)$ we have:

$$\beta(1) = p \circ \gamma(1) = p \circ \exp_e(\mathbf{u}) = \exp_p(DL_p(e)\mathbf{u}) = \exp_p(\mathbf{u}_p)$$

where the third equality comes from the change of base formulas for affine exponential 2.7. Following the same deduction and from the linearity of the differential, we have, for any $t \in [0, 1]$:

$$\beta(t) = p \circ \gamma(t) = p \circ \exp_e(t\mathbf{u}) = \exp_p(tDL_p(e)\mathbf{u}) = \exp_p(t\mathbf{u}_p)$$

□

**Lemma 2.3.1.** Let $\mathbb{G}$ be a finite dimensional connected Lie group, and $p, q, r$ three of its elements. If exists an $\epsilon$ such that

$$||\log(p \circ q) - \log(r)|| < \epsilon$$

then

$$||\log(p) - \log(q^{-1} \circ r)|| < \epsilon$$

Intuitively, the lemma states that if $p \circ q \simeq r$ then $p \simeq q^{-1} \circ r$.

The following theorem is an application of the pole ladder [LAP11] for the computation of the exponential that will underpin one of the numerical methods for the computation of the log-composition.

**Theorem 2.3.1.** Let $\mathbb{G}$ be a finite dimensional connected Lie group defined with a Cartan connection $\nabla$. Given two vectors $\mathbf{u}, \mathbf{v}$ in $\mathfrak{g}$, such that $p = \exp_e(\mathbf{u})$ and $q = \exp_e(\mathbf{v})$, with $\alpha$ integral curve of $\mathbf{u}$,

$$\alpha : [0,1] \to \mathbb{G} \quad \alpha(0) = e \quad \alpha(1) = p \quad \dot{\alpha}(0) = \mathbf{u}$$

and for $\mathbf{v}_p^{\parallel}$ parallel transport of $\mathbf{v}$

$$\mathbf{v}_p^{\parallel} = \Pi(\alpha)_e^p(\mathbf{v})$$

and $\mathbf{v}_e^{\parallel}$ pull-back of the left translation of the previous vector

$$\mathbf{v}_e^{\parallel} := DL_{p^{-1}}(e)(\mathbf{v}_p^{\parallel})$$

it follows that:

$$||\log_e(\exp_e(\mathbf{v}_e^{\parallel})) - \log_e(\exp_e\left(\frac{\mathbf{u}}{2}\right) \circ \exp_e(\mathbf{v}) \circ \exp_e\left(-\frac{\mathbf{u}}{2}\right))|| \leq |||[\mathbf{u}, \mathbf{v}]|||$$

The statement of this fairly intricate theorem involves a construction that can be visualized in figure 2.3, and will lead to a numerical method for the numerical computation of the Lie log-composition.

*Proof.* Let $m \in \mathbb{G}$ be the midpoint of the curve $\alpha$, $m = \alpha(1/2) = \exp_e(\frac{\mathbf{u}}{2})$ and let $\gamma$ be the geodesic between $q = \exp_e(\mathbf{v})$ and $m$:

$$\gamma(0) = q \qquad \gamma(1) = m \qquad \nabla_{\dot{\gamma}}\dot{\gamma} = 0$$

If $\mathbf{w}$ is the tangent vector of $\gamma$ defined at $q$ such that $\dot{\gamma}(0) = \mathbf{w}$, it follows from the change of base formula 2.7 that

$$\gamma(t) = \exp_q(t\mathbf{w}) = q \circ \exp_e(DL_{p^{-1}}(e)(t\mathbf{w})) = \exp_e(\mathbf{v}) \circ \exp_e(tDL_{p^{-1}}(e)\mathbf{w})$$

And by construction, we can move from the identity $e$ to $m$ directly walking the geodesic $\alpha$ or passing through $q$. It follows that

$$\begin{aligned} \exp_q(\mathbf{w}) &= \exp_e(\frac{\mathbf{u}}{2}) \\ q \circ \exp_e(DL_{q^{-1}}(e)(\mathbf{w})) &= \exp_e(\frac{\mathbf{u}}{2}) \\ \exp_e(DL_{q^{-1}}(e)(\mathbf{w})) &= \exp_e(-\mathbf{v}) \circ \exp_e(\frac{\mathbf{u}}{2}) \end{aligned}$$

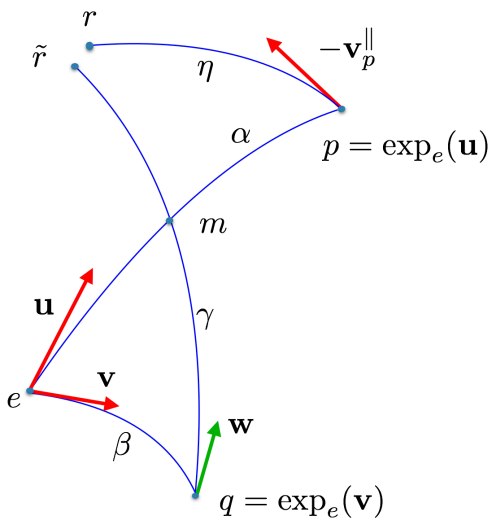



Figure 2.3: Pole ladder applied to parallel transport.

Let $\eta$ be the integral curve of the tangent vector $-\mathbf{v}_p^{\parallel}$ at $p$. We define two new points, $r := \eta(1)$ and $\tilde{r} := \gamma(2)$ where $\gamma$ is the integral curve of $2\mathbf{w}$.
On one side we have:

$$\begin{aligned}\tilde{r} = \gamma(2) &= \exp_q(2\mathbf{w}) = q \circ \exp_e(DL_{q^{-1}}(e)(2\mathbf{w})) \\ &= \exp_e(\mathbf{v}) \circ \exp_e(2DL_{q^{-1}}(e)\mathbf{w}) \\ &= \exp_e(\mathbf{v}) \circ \exp_e(DL_{q^{-1}}(e)\mathbf{w})^2 \\ &= \exp_e(\mathbf{v}) \circ \exp_e(\exp_e(-\mathbf{v}) \circ \exp_e(\frac{\mathbf{u}}{2}))^2 \\ &= \exp_e(\frac{\mathbf{u}}{2}) \circ \exp_e(-\mathbf{v}) \circ \exp_e(\frac{\mathbf{u}}{2})\end{aligned}$$

On the other side:

$$\begin{aligned}r = \eta(1) &= \exp_p(-\mathbf{v}_p^{\parallel}) = p \circ \exp_e(DL_{p^{-1}}(e)(-\mathbf{v}_p^{\parallel})) \\ &= \exp_e(\mathbf{u}) \circ \exp_e(-DL_{p^{-1}}(e)\mathbf{v}_p^{\parallel}) \\ &= \exp_e(\mathbf{u}) \circ \exp_e(-\mathbf{v}_e^{\parallel})\end{aligned}$$

having indicated $DL_{p^{-1}}(e)\mathbf{v}^{\parallel}$ with $\mathbf{v}_e^{\parallel}$ for brevity.
By geometrical construction, we have that if the space has no curvature (or equivalently, the Lie group is commutative), $r = \tilde{r}$. Therefore, using the change of signs property 2.1

$$\begin{aligned}\exp_e(\mathbf{u}) \circ \exp_e(-\mathbf{v}_e^{\parallel}) &= \exp_e(\frac{\mathbf{u}}{2}) \circ \exp_e(-\mathbf{v}) \circ \exp_e(\frac{\mathbf{u}}{2}) \\ \exp_e(\mathbf{v}_e^{\parallel}) &= \exp_e(\frac{\mathbf{u}}{2}) \circ \exp_e(\mathbf{v}) \circ \exp_e(-\frac{\mathbf{u}}{2})\end{aligned}$$

When the space is curved, again by construction, it follows that

$$||r - \tilde{r}|| \leq |||[\mathbf{u}, \mathbf{v}]|||$$

As a consequence of the previous lemma we have finally reached the thesis:

$$||\log_e(\exp_e(\mathbf{v}_e^{\parallel})) - \log_e(\exp_e\left(\frac{\mathbf{u}}{2}\right) \circ \exp_e(\mathbf{v}) \circ \exp_e\left(-\frac{\mathbf{u}}{2}\right))|| \leq |||[\mathbf{u}, \mathbf{v}]|||$$

□

The previous result can be reformulated as the approximation:

$$\exp_e(\mathbf{v}_e^{\parallel}) \simeq \exp_e\left(\frac{\mathbf{u}}{2}\right) \circ \exp_e(\mathbf{v}) \circ \exp_e\left(-\frac{\mathbf{u}}{2}\right) \tag{2.10}$$

that will turn out to be the main tool for the computation of the log-composition using parallel transport.

## 2.4 Numerical Computations of the Lie Log-composition

In this section we provide explicit formulas for the computation of the log composition:

$$\mathbf{v}_1 \oplus \mathbf{v}_2 = \log(\exp(\mathbf{v}_1) \circ \exp(\mathbf{v}_2)) \tag{2.11}$$

using the tools introduced in the previous sections.

### 2.4.1 Truncated BCH formula for the Lie Log-composition

As said in the end of section 2.2 the Lie log-composition possess a closed form, the BCH formula, defined as the solution of the equation $\exp(\mathbf{w}) = \exp(\mathbf{u}) \circ \exp(\mathbf{v})$, for $\mathbf{u}$ and $\mathbf{v}$ *analytic* elements in the Lie algebra $\mathfrak{g}$:

$$BCH(\mathbf{u},\mathbf{v}) = \mathbf{u} + \mathbf{v} + \frac{1}{2}[\mathbf{u},\mathbf{v}] + \frac{1}{12}([\mathbf{u},[\mathbf{u},\mathbf{v}]] + [\mathbf{v},[\mathbf{v},\mathbf{u}]]) - \frac{1}{24}[\mathbf{v},[\mathbf{u},[\mathbf{u},\mathbf{v}]]] + ... \tag{2.12}$$

It consists of an infinite series of Lie bracket whose asymptotic behaviour cannot be predicted only from the coefficient of each nested Lie bracket term. In practical applications it can be computed using its *approximation of degree $k$*, defined as the sum of the BCH terms having no more than $k$ nested Lie bracket. This convention is also coherent with the degree of the BCH expressed as polynomial formal series of adjoint operators (see next section 2.4.2):

$$\begin{aligned}
BCH^0(\mathbf{u},\mathbf{v}) &= \mathbf{u} + \mathbf{v} \\
BCH^1(\mathbf{u},\mathbf{v}) &= \mathbf{u} + \mathbf{v} + \frac{1}{2}[\mathbf{u},\mathbf{v}] \\
BCH^2(\mathbf{u},\mathbf{v}) &= \mathbf{u} + \mathbf{v} + \frac{1}{2}[\mathbf{u},\mathbf{v}] + \frac{1}{12}([\mathbf{u},[\mathbf{u},\mathbf{v}]] + [\mathbf{v},[\mathbf{v},\mathbf{u}]]) \\
BCH^3(\mathbf{u},\mathbf{v}) &= \mathbf{u} + \mathbf{v} + \frac{1}{2}[\mathbf{u},\mathbf{v}] + \frac{1}{12}([\mathbf{u},[\mathbf{u},\mathbf{v}]] + [\mathbf{v},[\mathbf{v},\mathbf{u}]]) - \frac{1}{24}[\mathbf{v},[\mathbf{u},[\mathbf{u},\mathbf{v}]]]
\end{aligned}$$

In numerical computations, nested Lie brackets can raise several issues, in particular when $\mathbf{u}$ and $\mathbf{v}$ are not close to the origin. Assuming that, as often happens for practical applications in imaging registration, $\mathbf{v}$ is smaller than $\mathbf{u}$, we define an intermediate degree for the truncated BCH formula, between 1 and 2:

$$BCH^{3/2}(\mathbf{u},\mathbf{v}) = \mathbf{u} + \mathbf{v} + \frac{1}{2}[\mathbf{u},\mathbf{v}] + \frac{1}{12}[\mathbf{u},[\mathbf{u},\mathbf{v}]]$$

Approximations of degree $k$ of the BCH formula can be considered as a first step toward the numerical approximations of the log-composition $\mathbf{u} \oplus \mathbf{v}$. The first limitation of truncating the BCH, as stated in section 1.2 is that there is no information on the error carried by each term.

Furthermore, they only apply to cases where $\mathbf{u}$ and $\mathbf{v}$ are analytic, so when we can expressed locally with a convergent power series, as in the case of tangent vectors to matrix Lie group. Additional limitations of this approximation can be found when applied to stationary velocity fields. This will be one of the topic of section 3.2.5.

### 2.4.2 Taylor Expansion Method for the Log-composition

A more sophisticated numerical method to manage the nested Lie brackets for the computation of the log-composition is based on the Taylor expansion.
As shown in the appendix of [KO89] the terms of the BCH can be recollected using the Hausdorff method: each of the therms containing the n-th power of the vector $\mathbf{v}$ are collected together in the formal series $A^n$. Therefore

$$BCH(\mathbf{u},\mathbf{v}) = \mathbf{u} + A^1\mathbf{v} + A^2\mathbf{v} + A^3\mathbf{v} + \cdots$$

Given the adjoint map:

$$\begin{aligned} ad_{\mathbf{u}} : \mathfrak{g} &\longrightarrow \mathfrak{g} \\ \mathbf{v} &\longmapsto ad_{\mathbf{u}}\mathbf{v} := [\mathbf{u},\mathbf{v}] \end{aligned}$$

and the multiple adjoint maps, defined as:

$$\mathrm{ad}^n_{\mathbf{u}}\mathbf{v} := [\underbrace{\mathbf{u},[\mathbf{u},...[\mathbf{u}}_{\text{n-times}},\mathbf{v}]...]]$$

$$\mathrm{ad}^{-n}_{\mathbf{u}}\mathbf{v} := [[...[\mathbf{v},\underbrace{\mathbf{u}]...],\mathbf{u}}_{\text{n-times}}] = (-1)^n\mathrm{ad}^n_{\mathbf{u}}\mathbf{v}$$

it can be demonstrated that then the operator $A^1$, when applied to $\mathbf{v}$ provides the linear part of $\mathbf{v}$ in the BCH formula and can be written as:

$$A^1 = \frac{\mathrm{ad}^{-1}_{\mathbf{u}}}{\exp(\mathrm{ad}_{\mathbf{u}}) - 1} = \sum_{n=0}^{\infty}\frac{(-1)^n B_n}{n!}\mathrm{ad}^{-n}_{\mathbf{u}} = \sum_{n=0}^{\infty}\frac{B_n}{n!}\mathrm{ad}^n_{\mathbf{u}}$$

where $\{B_n\}_{n=0}^{\infty}$ is the sequence of the second-kind Bernoulli number. If the first-kind Bernoulli number are used, then each term of the summation must be multiplied for $(-1)^n$, as did for example in [KO89]. The denominator is defined within the structure of the formal power series ring [MT13].
In conclusion, the log-composition can be expressed as:

$$\mathbf{u}\oplus\mathbf{v} = \mathbf{u} + \frac{\mathrm{ad}^{-1}_{\mathbf{u}}}{\exp(\mathrm{ad}_{\mathbf{u}}) - 1}\mathbf{v} + \mathcal{O}(\mathbf{v}^2)$$

$$\mathbf{u}\oplus\mathbf{v} = \mathbf{u} + \sum_{n=0}^{\infty}\frac{B_n}{n!}\mathrm{ad}^n_{\mathbf{u}}\mathbf{v} + \mathcal{O}(\mathbf{v}^2) \tag{2.13}$$

that will turn out to be an important tool for the computation of the log-composition in the finite dimensional case.

### 2.4.3 Parallel Transport Method for the Log-composition

To obtain a numerical computation for the log-composition using parallel transport, we have to consider the following two assumptions:

1. If $\mathbf{v}_e^{\parallel}$ is defined as in theorem 2.3.1, then

$$||\mathbf{u}\oplus\mathbf{v} - (\mathbf{u} + \mathbf{v}_e^{\parallel})|| \leq ||[\mathbf{u},\mathbf{v}]||$$

2. If the vector $\mathbf{u} \in \mathfrak{g}$ is small enough, then:

$$\exp(\mathbf{u}) \simeq e + \mathbf{u}$$

The first assumption is a consequence of geometrical intuition. On a flat space, or a space with no curvature, the geodesics are straight lines, and $\mathbf{u} \oplus \mathbf{v} = \mathbf{u} + \mathbf{v}$ that is equal, again intuitively, to the sum of $\mathbf{u}$ with the parallel transported of $\mathbf{v}$ to the point $\exp_e(\mathbf{u})$, indicated with $\mathbf{v}_p^{\parallel}$ (see figure 2.4). It is not possible to sum two vectors belonging to two different planes, therefore we have to consider the transported $\mathbf{v}_e^{\parallel}$ instead of $\mathbf{v}_p^{\parallel}$. In addition, when the space is not flat, the equalities $\mathbf{u} \oplus \mathbf{v} = \mathbf{u} + \mathbf{v}$ do not holds.

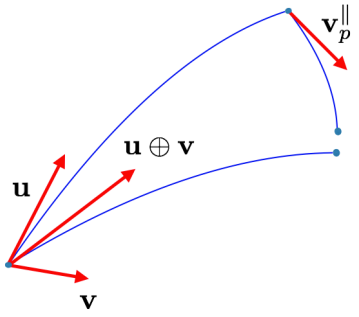



Figure 2.4: Representation of the intuitive idea of the computation of the log-composition using parallel transport. When the space is curved: $\mathbf{u} + \mathbf{v} \neq \mathbf{u} \oplus \mathbf{v} \simeq \mathbf{u} + \mathbf{v}_e^{\parallel}$ .

The validity of the second assumption must be investigated case by case. For example when $\mathbb{G}$ is a matrix Lie group, the formula 2.2 provides $\exp(\mathbf{u}) = I + \mathbf{u} + \mathcal{O}(\mathbf{u}^2)$. In the case of stationary velocity fields, the condition holds when $\mathbf{u}$ is small enough (see proposition 8.6 pag. 163 [You10]). More on this will be presented in 3.2.5.

Assuming the validity of these assumptions and from equation 2.10 it follows that

$$\begin{aligned} \mathbf{u} \oplus \mathbf{v} &\simeq \mathbf{u} + \mathbf{v}_e^{\parallel} \\ e + \mathbf{v}_e^{\parallel} &\simeq \exp_e\left(\frac{\mathbf{u}}{2}\right) \circ \exp_e(\mathbf{v}) \circ \exp_e\left(-\frac{\mathbf{u}}{2}\right) \end{aligned}$$

Therefore

$$\mathbf{u} \oplus \mathbf{v} \simeq \mathbf{u} + \exp_e\left(\frac{\mathbf{u}}{2}\right) \circ \exp_e(\mathbf{v}) \circ \exp_e\left(-\frac{\mathbf{u}}{2}\right) - e \tag{2.14}$$

Another important assumption we are making when the previous formula is applied to stationary velocity fields is that it holds also when the Lie group is infinite dimensional. An eventual confirmation of this is at the moment not known to the author. We assume it is true in coherence with what has been said in the introduction, section 1.4 about a naive approach to the infinite dimensional Lie Group theory.

With the truncated BCH and the Taylor expansion, equation 2.14 is the third numerical method for the computation of the log-composition explored in this thesis. The next chapter is devoted to introduce two groups of transformation - the rigid body transformations and the diffeomorphisms - and to apply the numerical methods presented in this chapter to these cases.

# Chapter 3

# Spatial Transformations Considered for the Computations of the Log-composition: $SE(2)$ and Diff$(\Omega)$

In the previous chapter we have introduced some mathematical tools for the numerical computation of the log-composition. Each of the theoretical elements depends strongly on the transformations considered, and in this chapter we will see how they can be applied for the transformations belonging to $SE(2)$ and Diff$(\Omega)$.

## 3.1 The Lie Group of Rigid Body Transformations

Each element of the group of rigid body transformation (or euclidean group) $SE(2)$ can be computed as a consecutive application of a rotation and a translation applied to any point $(x, y)^T$ of the plane:

$$\begin{pmatrix} X \\ Y \end{pmatrix} = R(\theta) \begin{pmatrix} x \\ y \end{pmatrix} + t = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t^x \\ t^y \end{pmatrix}$$

where the rotation matrix indicated with $R(\theta)$ belongs to the special orthogonal group $SO(2)$ and the translation $t$ is a vector of the plane.
We can represent the elements of $SE(2)$ in two different form: as ternary vector (restricted form)

$$SE(2)^v := \{(\theta, t^x, t^y) \mid \theta \in [0, 2\pi), t^x, t^y \in \mathbf{R}^2\}$$

or with matrices (matrix form)

$$SE(2) := \left\{ \begin{pmatrix} R(\theta) & t \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) & t^x \\ \sin(\theta) & \cos(\theta) & t^y \\ 0 & 0 & 1 \end{pmatrix} \mid \theta \in [0, 2\pi), (t^x, t^y) \in \mathbf{R}^2 \right\}$$

The group $SE(2)$ it is a manifold with a differentiable structure compatible with the operation of composition, whose Lie algebra is given in matrix form by

$$\mathfrak{se}(2) := \left\{ \begin{pmatrix} dR(\theta) & dt \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & -\theta & dt^x \\ \theta & 0 & dt^y \\ 0 & 0 & 0 \end{pmatrix} \mid \theta \in [0, 2\pi), (dt^x, dt^y) \in \mathbf{R}^2 \right\}$$

and it is indicated with $\mathfrak{se}(2)^v$ in its restricted form.

Given $r$, element of $SE(2)$ with $\theta \neq 0$, its image with the Lie group logarithm is

$$\begin{aligned} \log(r) &= \sum_{k=1}^{\infty} (-1)^{k+1} \frac{(r-I)^k}{k} = \begin{pmatrix} dR(\theta) & L(\theta)t \\ 0 & 1 \end{pmatrix} \\ &= \begin{pmatrix} 0 & -\theta & \frac{\theta}{2}(\frac{\sin(\theta)}{1-\cos(\theta)}t^x + t^y) \\ \theta & 0 & \frac{\theta}{2}(-t^x + \frac{\sin(\theta)}{1-\cos(\theta)}t^y) \\ 0 & 0 & 0 \end{pmatrix} \end{aligned}$$

where

$$dR(\theta) = \begin{pmatrix} 0 & -\theta \\ \theta & 0 \end{pmatrix} \qquad L(\theta) = \frac{\theta}{2} \begin{pmatrix} \frac{\sin(\theta)}{1-\cos(\theta)} & 1 \\ -1 & \frac{\sin(\theta)}{1-\cos(\theta)} \end{pmatrix}$$

On the way back, the exponential of $dr \in \mathfrak{se}(2)$ is given by:

$$\begin{aligned} \exp(dr) &= \sum_{k=1}^{\infty} \frac{dr^k}{k!} = \begin{pmatrix} R(\theta) & L(\theta)^{-1}dt \\ 0 & 1 \end{pmatrix} \\ &= \begin{pmatrix} \cos(\theta) & -\sin(\theta) & \frac{1}{\theta}(\sin(\theta)dt^x - (1-\cos(\theta))dt^y) \\ \sin(\theta) & \cos(\theta) & \frac{1}{\theta}(-(1-\cos(\theta))dt^x + \sin(\theta)dt^y) \\ 0 & 0 & 1 \end{pmatrix} \end{aligned}$$

where

$$L(\theta)^{-1} = \frac{1}{\theta} \begin{pmatrix} \sin(\theta) & -(1-\cos(\theta)) \\ (1-\cos(\theta)) & \sin(\theta) \end{pmatrix}$$

When $\theta$ is zero, $R(\theta)$ and $dR(\theta)$ coincide with the identity, and the transformation results in a translation. For proof and further details see for example [Gal11] [Hal15].

At this point it is important to notice that:

1. The infinite series of matrices of equation 2.2 does not raises any theoretical issue. An element of a group can be considered as a sum, where this operation is defined in the bigger algebra that contains both the Lie group and the Lie algebra. It appears to be the natural way to move back and forth from the group to the algebra. A second door to pass from one structure to the other, when the rotation $\theta$ is small is provided by the following approximations:

$$\exp(r) \simeq I + r \qquad \log(dr) \simeq dr - I \tag{3.1}$$

   In fact for small $\theta$, $\sin(\theta) \simeq \theta$, $\cos(\theta) \simeq 1$ and $L(\theta) \simeq I$.

2. The map exp is not well defined as bijection over its whole domain $\mathfrak{se}(2)$. Given two elements $(\theta_0, dt_0^x, dt_0^y)$ and $(\theta_1, dt_1^x, dt_1^y)$, they have the same image with exp function if the two following conditions are both satisfied:

   i) Exists an integer $k$ such that $\theta_0 = \theta_1 + 2k\pi$.

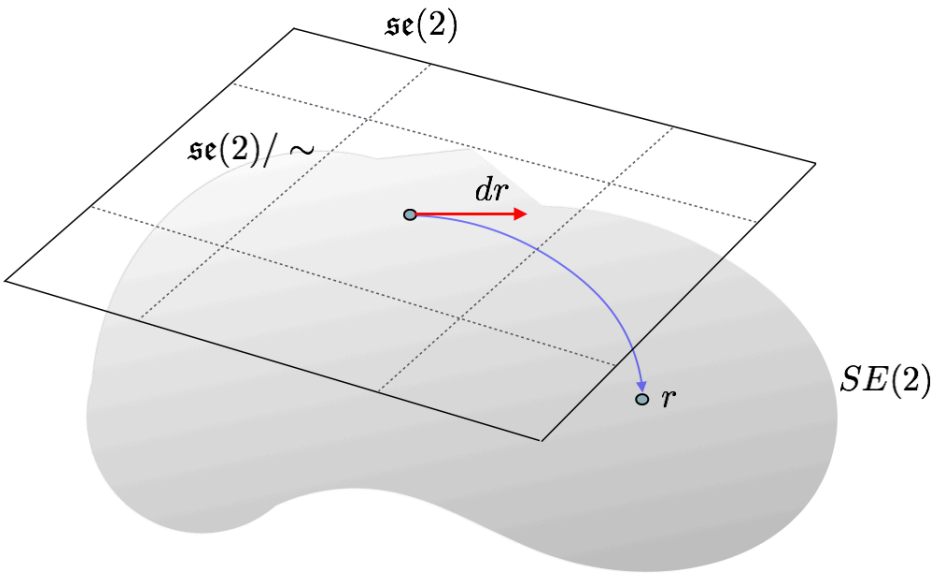


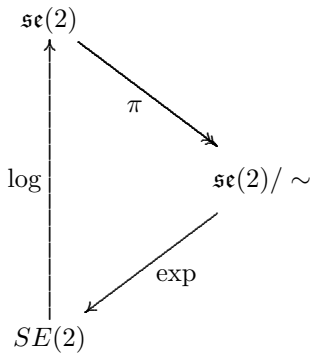

Figure 3.1: The Lie algebra $\mathfrak{se}(2)/\sim$ defined as the quotient of the Lie algebra $\mathfrak{se}(2)$ over the equivalence relation $\sim$ is in bijective correspondence with $SE(2)$.

ii) When $\theta_0 \neq 0$ the translation $(dt_0^x, dt_0^y)$ coincides with $(dt_1^x, dt_1^y)$ up to a factor $\frac{\theta_0}{\theta_1}$, where the angles are considered modulo $2\pi$.

To have a bijective correspondence the domain of exp has to be restricted to a space where if $\exp(\theta_0, dt_0^x, dt_0^y) = \exp(\theta_1, dt_1^x, dt_1^y)$ implies $(\theta_0, dt_0^x, dt_0^y) = (\theta_1, dt_1^x, dt_1^y)$. It can be easy to prove that the sought space is the quotient of $\mathfrak{se}(2)$ over the equivalence relation $\sim$, defined as

$$\begin{aligned} (\theta_0, dt_0^x, dt_0^y) &\sim (\theta_1, dt_1^x, dt_1^y) \\ &\iff \text{(by definition)} \\ \exists k \in \mathbb{Z} \mid \theta_0 = \theta_1 + 2k\pi \quad &\text{and} \quad (dt_0^x, dt_0^y) = \frac{\theta_0}{\theta_1}(dt_1^x, dt_1^y) \end{aligned}$$

The new algebra defined by the set of equivalence classes of this relation is indicated - with the standard convention, see [Art11] - with $\mathfrak{se}(2)/\sim$. With this restriction of the domain, the function exp is a bijection having log as its inverse. What said so far can be summarize in the following commutative diagram, where $\pi$ is the canonical projection:




and with the schematic figure 3.1.

### 3.1.1 Computations of Log-composition in $\mathfrak{se}(2)$

The log-composition of two elements $dr_0 = (\theta_0, dt_0^x, dt_0^y)$ ans $dr_1 = (\theta_1, dt_1^x, dt_1^y)$ of $\mathfrak{se}(2)/\sim$ results

$$dr_0 \oplus dr_1 = \log(\exp(dr_0) \circ \exp(dr_1)) \tag{3.2}$$

The approximations of the log-composition using truncated BCH formulas are straightforward:

$$dr_0 \oplus dr_1 \simeq BCH^0(dr_0, dr_1) := dr_0 + dr_1$$
$$dr_0 \oplus dr_1 \simeq BCH^1(dr_0, dr_1) := dr_0 + dr_1 + \frac{1}{2}[dr_0, dr_1]$$
$$dr_0 \oplus dr_1 \simeq BCH^{3/2}(dr_0, dr_1) := dr_0 + dr_1 + \frac{1}{2}[dr_0, dr_1] + \frac{1}{12}[dr_0, [dr_0, dr_1]]$$
$$dr_0 \oplus dr_1 \simeq BCH^2(dr_0, dr_1) := dr_0 + dr_1 + \frac{1}{2}[dr_0, dr_1] + \frac{1}{12}([dr_0, [dr_0, dr_1]] + [dr_1, [dr_1, dr_0]])$$

To compute the approximation with the Taylor method, and so to compute the equation 2.13 for elements in $\mathfrak{se}(2)/\sim$, we observe that the restricted form of the Lie bracket is given by

$$\begin{aligned}[dr_0, dr_1] &= (0, dR(\theta_0)dt_1 - dR(\theta_1)dt_0)^T \\ &= (0, -\theta_0 dt_1^y + \theta_1 dt_0^y, \theta_0 dt_1^x - \theta_1 dt_0^x)^T\end{aligned}$$

Therefore, the adjoint operator can be written in matrix form as a dual matrix of $dr$:

$$\mathrm{ad}_{dr} = \begin{pmatrix} 0 & 0 & 0 \\ dt^y & 0 & -\theta \\ -dt^x & \theta & 0 \end{pmatrix}$$

In fact, when applied to $dr_1$ it results in the Lie bracket:

$$\mathrm{ad}_{dr_0} dr_1 = \begin{pmatrix} 0 & 0 & 0 \\ dt_0^y & 0 & -\theta_0 \\ -dt_0^x & \theta_0 & 0 \end{pmatrix} \begin{pmatrix} \theta_1 \\ dt_1^x \\ dt_1^y \end{pmatrix} = \begin{pmatrix} 0 \\ -\theta_0 dt_1^y + \theta_1 dt_0^y \\ \theta_0 dt_1^x - \theta_1 dt_0^x \end{pmatrix}$$

To compute the Taylor approximation proposed in equation 2.13 of the log composition, indicating $dt^\star = (dt^y, -dt^x)$ it can be proved easily by induction that

$$\mathrm{ad}_{dr}^n = \begin{pmatrix} 0 & 0 \\ dt^\star & dR(\theta) \end{pmatrix}^n = \begin{pmatrix} 0 & 0 \\ dR(\theta)^{n-1} dt^\star & dR(\theta)^n \end{pmatrix}$$

And so the series involved in the equation 2.13 become

$$\sum_{n=0}^{\infty} \frac{B_n}{n!} \mathrm{ad}_{dr}^n = \sum_{n=0}^{\infty} \frac{B_n}{n!} \begin{pmatrix} 0 & 0 \\ dR(\theta)^{n-1} dt^\star & dR(\theta)^n \end{pmatrix}$$

We can split it in two part, the rotational part $dR(\theta)^n$ and the translational part $dR(\theta)^{n-1}dt^\star$. The rotational part, exploiting the nature of Bernoulli numbers and its generative equation, when $\theta \neq 0$, becomes

$$\begin{aligned}
\sum_{n=0}^{\infty} \frac{B_n}{n!} dR(\theta)^n &= I + \frac{1}{2} dR(\theta) + \sum_{n=1}^{\infty} \frac{B_{2n}}{2n!} dR(\theta)^{2n} \\
&= I + \frac{1}{2} dR(\theta) + (\sum_{n=1}^{\infty} \frac{B_{2n}}{2n!} (i\theta)^{2n}) I \\
&= \frac{1}{2} dR(\theta) + (\sum_{n=0}^{\infty} \frac{B_n}{n!} (i\theta)^n - \frac{1}{2} i\theta) I \\
&= \frac{1}{2} dR(\theta) + (\frac{i\theta e^{i\theta}}{e^{i\theta} - 1} - \frac{1}{2} i\theta) I \\
&= \frac{1}{2} dR(\theta) + \frac{\theta/2}{\tan(\theta/2)} I
\end{aligned}$$

where the equation $dR(\theta)^{2n} = (i\theta)^{2n} I$. For the translational part we have

$$\begin{aligned}
\sum_{n=1}^{\infty} \frac{B_n}{n!} dR(\theta)^{n-1} dt^\star &= dR(\theta)^{-1} \Big( \sum_{n=1}^{\infty} \frac{B_n}{n!} dR(\theta)^n \Big) dt^\star \\
&= dR(\theta)^{-1} \Big( \sum_{n=0}^{\infty} \frac{B_n}{n!} dR(\theta)^n - I \Big) dt^\star \\
&= dR(\theta)^{-1} \Big( \sum_{n=0}^{\infty} \frac{1}{2} dR(\theta) + \frac{\theta/2}{\tan(\theta/2)} I - I \Big) dt^\star \\
&= dR(\theta)^{-1} \Big( \sum_{n=0}^{\infty} \frac{1}{2} dR(\theta) + \frac{\theta/2}{\tan(\theta/2)} I - I \Big) dt^\star \\
&= \Big( \frac{1}{2} I + (\frac{\theta/2}{\tan(\theta/2)} - 1) dR(\theta)^{-1} \Big) dt^\star
\end{aligned}$$

Finally the closed form for the Taylor approximation of the log-composition is [Ver14]:

$$dr_0 \oplus dr_1 = dr_0 + \sum_{n=0}^{\infty} \frac{B_n}{n!} \mathrm{ad}_{dr_0}^n dr_1 + \mathcal{O}(dr_1^2) = dr_0 + \mathbf{J}(dr_0) dr_1 + \mathcal{O}(dr_1^2) \tag{3.3}$$

where

$$\mathbf{J}(dr_0) = \begin{pmatrix} 1 & 0 & 0 \\ -\frac{\theta_0/2 - \tan(\theta_0/2)}{\theta_0 \tan(\theta_0/2)} dt_0^x + \frac{1}{2} dt_0^y & \frac{\theta_0/2}{\tan(\theta_0/2)} & -\theta_0/2 \\ -\frac{1}{2} dt_0^x - \frac{\theta_0/2 - \tan(\theta_0/2)}{\theta_0 \tan(\theta_0/2)} dt_0^y & \theta_0/2 & \frac{\theta_0/2}{\tan(\theta_0/2)} \end{pmatrix}$$

therefore the corresponding numerical method indicated with the function Tl as

$$dr_0 \oplus dr_1 \simeq Tl(dr_0, dr_1) := dr_0 + \mathbf{J}(dr_0) dr_1 \tag{3.4}$$

The approximation of the log-composition using parallel transport is a straightforward application of the equation 2.14:

$$dr_0 \oplus dr_1 \simeq pt(dr_0, dr_1) := dr_0 + \exp\left(\frac{dr_0}{2}\right) \exp(dr_1) \exp\left(-\frac{dr_0}{2}\right) - I \tag{3.5}$$

where the composition in the Lie group coincides with the product of matrix in the bigger algebra $GL(3)$ that contains both the Lie group $SE(2)$ and the Lie algebra $\mathfrak{se}(2)$.

## 3.2 A Lie Group Structure for Diffeomorphisms

The passage from the finite to the infinite dimensional case is not free of deceptions. We will investigate in the next two subsections, 3.2.1 and 3.2.2 ,the following facts that are true matrices but not for diffeomorphisms:

1. Lie logarithm and Lie exponential are local isomorphisms.
2. $SE(2)$ and $\mathfrak{se}(2)$ are subset of a bigger algebra, where all of the operations are compatible.

To avoid ambiguities, we first need to write down some definitions and facts.

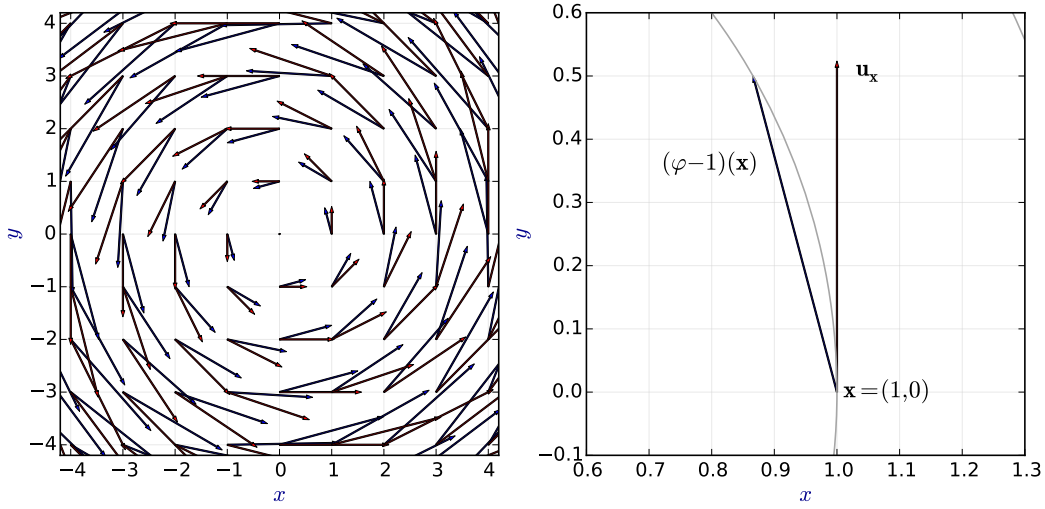



Figure 3.2: The displacement field and the tangent vector field for the transformation $\varphi$ defined as a rotation of $\pi/6$ around the origin. When $\varphi$ is subtracted by the identity function, it becomes an element of the algebra of the velocity vector fields Vect($\Omega$).

The set of *deformations* is the set of continuous functions from $\Omega$ to $\Omega$, compact subset of $\mathbb{R}^d$. If a deformation is invertible with continuous inverse, then it is called *homeomorphism*; the set of homeomorphisms forms a group, indicated with Hom($\Omega$), with the operation of function composition. If an homeomorphism is differentiable and has differentiable inverse then it is called *diffeomorphism*. Again the set of diffeomorphisms forms a group, indicated with Diff($\Omega$).

A *velocity vector field* over $\Omega$ is a differentiable function that at each point of $\Omega$, associates a vector of $\mathbb{R}^d$; the set of velocity vector fields, indicated with Vect($\Omega$) forms a vector space, and considering the Lie bracket defined by the directional derivative we obtain that Vect($\Omega$)

forms a Lie algebra[1]. If $\{\frac{\partial}{\partial x_i}\}_{i=1}^{d}$ is a local coordinates system over $\Omega$, $\mathbf{u} = a^i \frac{\partial}{\partial x_i}$ and $\mathbf{v} = b^i \frac{\partial}{\partial x_i}$ are two elements of Vect$(\Omega)$ written using the Einstein summation convention, the Lie bracket can be expressed using the Jacobian:

$$\begin{aligned}
[\mathbf{u}, \mathbf{v}] &= a^i \frac{\partial}{\partial x_i}\big(b^j \frac{\partial}{\partial x_j}\big) - b^i \frac{\partial}{\partial x_i}\big(a^j \frac{\partial}{\partial x_j}\big) \\
&= a^i \frac{\partial b_j}{\partial x_i}\frac{\partial}{\partial x_j} + a^i b^j \frac{\partial^2}{\partial x_i x_j} - b^j \frac{\partial a^i}{\partial x^j}\frac{\partial}{\partial x_i} - b^j a^i \frac{\partial^2}{\partial x_j x_i} \\
&= a^i \frac{\partial b_j}{\partial x_i}\frac{\partial}{\partial x_j} - b^j \frac{\partial a^i}{\partial x^j}\frac{\partial}{\partial x_i} \\
&= J_{\mathbf{v}}\mathbf{u} - J_{\mathbf{u}}\mathbf{v}
\end{aligned}$$

It was proved that the Lie algebra of the Lie group of diffeomorphisms Diff$(\Omega)$ is Vect$(\Omega)$ ([Mil82], [OKC92]). Therefore a single vector in the Lie algebra Vect$(\Omega)$ (represented by one red arrow in figure 1.1) is a vector field defined over $\Omega$ (represented by the set of red arrows in figure 3.2). We would expect that to a single diffeomorphism in the Lie group corresponds a single vector of the Lie algebra. This is not the case, since Lie logarithm and Lie exponential are not local isomorphisms on the whole domain.

### 3.2.1 Local isomorphisms for a subset of Diffeomorphisms: one-parameter subgroup and stationary velocity fields

In the case of matrices, the exponential map is a local isomorphisms: it is always possible to find an open neighbor of $\mathbf{0}$ in the Lie algebra and an open neighbor of the identity element in the Lie group (in the same topology induced by the metric inherited by the bigger algebra), such that the exponential map is defined and invertible.

In the infinite dimensional case there are diffeomorphisms arbitrarily close to the identity that are not embedded into any one-parameter subgroups, therefore the exponential map is not a local isomorphism (see the counterexample in [Mil84b], pag. 1017 or the definition of Koppel-diffeomorphisms [Gra88] pag. 115).

Since for medical image registration we are interested only in the diffeomorphisms that can be parametrized by tangent vector fields, this feature is worth being investigated, but it requires some definitions.

If $\varphi$ is a one-parameter subgroup on the manifold *Diff*$(\Omega)$, then its derivative satisfies the *stationary* (or homogeneous) ordinary differential equation:

$$\frac{d\varphi(t)}{dt} = V_{\varphi(t)} \tag{3.6}$$

where the stationary vector field $V_{\varphi(t)}$ defined over $\Omega$ is an element of the Lie algebra of $\mathcal{V}(\Omega)$ called *stationary velocity field* (SVF).

Vice versa, given an SVF, thanks to Cauchy theorem, exists always a unique solution $\varphi$ to the ordinary differential equation (ODE) 3.6, given the initial condition $\varphi(0) = 1$, that satisfies the property of one-parameter subgroup.

We indicate with *Diff*$^1(\Omega)$ the *set of diffeomorphisms embedded in a one parameter subgroup*, or the solutions of 3.6 for any differentiable vector field defined over $\Omega$ .

We can extend our attention to non-stationary (or non-homogeneous or non-autonomous) ordinary differential equation of the form:

$$\frac{d\psi(t)}{dt} = W_{(t,\psi(t))} \tag{3.7}$$

[1] Some books invert the signs of the operation (see the Kirillow's remarks [Kir08] pag. 27); this choice does not have any impact in the study of the algebraic structure, but it has an impact on the numerical results when the Lie brackets are implemented for numerical computations. At the moment the sign that defines the Lie bracket is chosen on the base of the obtained results.

where $W_{(t,\psi(t))}$ is a non-stationary vector field, called here time varying vector field, or TVVF.

When compared to the SVF, it does not depend only on the spatial position $\mathbf{x}$ but there is also a temporal dependency. Think for example to an artificial satellite orbiting around the globe: it is subject to the earth's vector field in respect to which it is constant for a fixed position, and to the lunar vector field that it is not fixed but varies with respect to the time. Conventionally the temporal domain $T$ contains the origin and formally we can write:

$$\begin{aligned} W : T \times \Omega &\longrightarrow \mathbb{R}^d \\ (t, \psi(t)) &\longmapsto W_{(t,\psi(t))} \end{aligned}$$

for $\psi$ diffeomorphism that, when applied to a point of $\Omega$ at the time $t$, is indicated with $\psi(t, \mathbf{x})$ or $\psi^{(t)}(\mathbf{x})$.

Non-stationary ODE are particular cases of a stationary ones. Writing the diffeomorphism $\psi(t)$ applied to $\mathbf{x}$ in local coordinates as

$$\psi^{(t)}(\mathbf{x}) = (\psi_1^{(t)}(\mathbf{x}), \psi_2^{(t)}(\mathbf{x}), \dots, \psi_d^{(t)}(\mathbf{x})) \in \mathbb{R}^d$$

and defining a new function $\psi_0^{(t)}(\mathbf{x}) = t_0 + t$ for all $\mathbf{x} \in \Omega$, we can obtain then the new diffeomorphism $\tilde{\psi}^{(t)}$ that in local coordinates is expressed as

$$\tilde{\psi}^{(t)}(\mathbf{x}) = (\psi_0^{(t)}(\mathbf{x}), \psi_1^{(t)}(\mathbf{x}), \psi_2^{(t)}(\mathbf{x}), \dots, \psi_d^{(t)}(\mathbf{x})) \in T \times \mathbb{R}^d$$

that reduces the ODE 3.7 to an ODE of the form 3.6. In the example of the satellite, is like considering the temporal dimension as an additional dimension of the space.

It follows that stationary ODE and non-stationary ODE have solutions that belong to $\mathit{Diff}^1(\Omega)$ and $\mathit{Diff}^1(T \times \Omega)$ respectively. For each instant of time the solutions of a non-stationary ODE are embedded in the set of one-parameter subgroup of $\mathit{Diff}(\Omega)$, but for two different instant of time, the solutions can belong to two different elements in the one parameter subgroups.

In conclusion, we have that in the case of diffeomorphisms exp is not a local isomorphism, unless we restrict the group of diffeomorphisms to the one embedded in a one parameter subgroup $\mathit{Diff}^1(T \times \Omega)$.

In the LDDMM framework [BMTY05] TVVFs are initially considered, while with the paper of Arsigny [ACPA06], and in subsequent works, the attention has been restricted to SVF. This is due to the fact that scaling and squaring algorithm and inverse scaling and squaring algorithm, as every numerical method based on the phase flow to compute the Lie exponential and the Lie logarithm, works only under the assumption that the transformation belongs to the same one-parameter subgroup.

### 3.2.2 A bigger algebra for the group of Diffeomorphisms

As well as for any matrix Lie group, both the group $SE(2)$ and the algebra $\mathfrak{se}(2)$ are subset of the same bigger algebra of matrices in the general linear group $GL(3, \mathbb{R})$. The product of the algebra coincides with the composition of the group and thanks to the linearity, scalar product is compatible both with the product and the composition.

The existence of a bigger algebra is not important only in the research of an elegant structure: the power series expansions of the exponential (2.2) and the logarithm (2.3) as well as expressions like (2.4) and (2.5) would be meaningless without the possibility of expressing the sum of two elements of a multiplicative group. Moreover, if the bigger algebra that contains both Lie group and Lie algebra exists, a unique norm in this space can be defined and utilized to compare elements in the both subspaces.

In the case of diffeomorphisms of the compact subset $\Omega$ of $\mathbb{R}^d$, we can not identify a bigger vector space that contains both Lie group and Lie algebra, but we can reach something similar

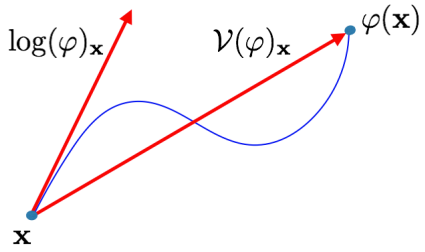



Figure 3.3: For small deformations, the displacement field $\log(\varphi)$ and the tangent field $\mathcal{V}(\varphi)$, computed at the point $\mathbf{x}$ of $\Omega$, are close to each others.

than what happens for matrices mapping diffeomorphisms to velocity vector fields over $\Omega$. For this aim it is necessary to have some definitions at hand.

There are two ways of associating a diffeomorphism $\varphi$ to a velocity vector field. The first one is elementary but fundamental in this context: it consists in subtracting to $\varphi$ the identity function 1. If $\mathbf{x}$ is in $\Omega$ and $\varphi(\mathbf{x})$ is the new point after the transformation, then the associated velocity vector field, called here *displacement field of* $\varphi$, is the function that at the point $\mathbf{x}$ associates the vector defined as the difference $\varphi(\mathbf{x}) - \mathbf{x}$. We indicate this operation of adding and subtracting the identity with the function $\mathcal{V}$:

$$\mathcal{V}(\varphi) = \varphi - 1 \qquad\qquad \mathcal{V}^{-1}(\mathbf{u}) = \mathbf{u} + 1$$

where $\mathbf{u} + 1$ is called *deformation of* $\mathbf{u}$.

We can see that displacement fields of diffeomorphisms are elements of $\mathrm{Vect}(\Omega)$, the infinite dimensional vector space defined by the vector field over $\Omega$ that plays a similar role to the bigger algebra for matrices. This infinite dimensional vector space is a Lie algebra if considered with the Lie bracket and in particular is the Lie algebra of the Lie group of diffeomorphisms (see [KW08]). Since it contains also the diffeomorphisms when expressed in the form $\mathcal{V}^{-1}(\mathbf{u})$ it plays the same role of the bigger algebra that contains Lie group and Lie algebra in the case of matrices. We can observe that this operation of subtracting the identity to the deformation has already been used implicitly in the power series expansion of the Lie logarithm for matrices, see equation 2.3.

The second way to associate a velocity vector field to $\varphi$ is to use the Lie logarithm defined in chapter 2. It is interesting to notice that when $\mathbf{u}$ is small then $\mathcal{V}^{-1}$ and exp are closed to each other and $\mathcal{V}^{-1}$ can be considered a good approximation of exp (see figure 3.3). The very same happens for matrices, as seen in equations (3.1).

### 3.2.3 A Norm for the Elements in the one-parameter subgroup

A metric between tangent vector fields of $\Omega$ can be defined as

$$d(\mathbf{u}, \mathbf{v}) = \Big( \int_{\Omega} ||\mathbf{u} - \mathbf{v}||^2_{L^2} d\mathbf{x} \Big)^{1/2} \tag{3.8}$$

that naturally induces a metric on the Lie algebra.

The Lie group $Diff^1(\Omega)$ does not possess any norm, but the corresponding displacement fields defined using $\mathcal{V}$, as element of $\mathrm{Vect}(\Omega)$ can be considered with a norm. Given two

diffeomorphisms $\varphi_0$ and $\varphi_1$ we have

$$d^1(\varphi_0, \varphi_1) = \Big( \int_\Omega ||\mathcal{V}(\varphi_0) - \mathcal{V}(\varphi_1)||^2_{L^2} d\mathbf{x} \Big)^{1/2} \tag{3.9}$$

Despite the limitation that Lie algebra and Lie group of diffeomorphisms are not subset of any bigger algebra, thanks to $\mathcal{V}$ we can nevertheless consider a function that measures the distance between $Diff^1(\Omega)$ and SVF:

$$g(\mathbf{u}, \varphi) = \Big( \int_\Omega ||\mathbf{u} - \mathcal{V}(\varphi)||^2_{L^2} d\mathbf{x} \Big)^{1/2} \tag{3.10}$$

The next section shows the parametrization of SVF utilized in the applications. It is followed by the presentation of the numerical methods employed for the computation of the Lie log-composition applied to SVF.

### 3.2.4 Parametrization of SVF: Grids and Discretized Vector Fields

Even if images are discrete objects, we model them as continuous functions. There are several motivations that led to this choice: as underlined by [Sze94], images are discrete measurement of the continuous property of an object. Therefore it is reasonable to have a model as close as possible to the continuous object rather than to a set of discrete measurements. Certainly it is important to keep in mind the fact that the continuous approximation is obtained - in a non unique way - from the discretized image with an interpolation scheme. This implies that, if the distance between two separate objects is less than the size of a unitary element (pixel/voxel), in a continuous approximation based on the discretized image the two objects objects will not be separated anymore.

Furthermore transformations between images are discretized vector fields, where each vector is applied to an element of a grid. These transformations can only be considered as a model of the group of diffeomorphisms (a model of a model, in image registration!) and reflects only partially the continuous property of the original transformation. In turn, the use of discretized elements allows for the computerisation of the solutions.

The data structure utilized to store images, as well as displacement fields, is a 5-dimensional matrix

$$M = M(x_i, y_j, z_k, t, d) \qquad (i, j, k) \in L, \quad t \in T \quad d = 1, 2, 3 \tag{3.11}$$

where $(x_i, y_j, z_k)$ are the discrete positions on a lattice $L$ in the domain of the images, $t$ is the time parameter in a discretized domain $T$ and $d$ is the index of the coordinate axis. So, the discretized *tangent vector* $\mathbf{v}_\tau(x_i, y_j, z_k)$ at time $t$, has coordinates defined by

$$\mathbf{v}_t(x_i, y_j, z_k) = (M(x_i, y_j, z_k, t, 1), M(x_i, y_j, z_k, t, 2), M(x_i, y_j, z_k, t, 3))$$

### 3.2.5 Computations of Log-composition for SVF

The Lie bracket that appears in the truncated $BCH$ of degree $0, 1, 1.5$ and $2$ for SVF, are computed using the Jacobian matrix $J$:

$$[\mathbf{u}, \mathbf{v}] := \mathbf{v} \cdot \mathbf{v} - \mathbf{v} \cdot \mathbf{u} = J_u \mathbf{v} - J_v \mathbf{u} \qquad \forall \mathbf{u}, \mathbf{v} \in \mathfrak{g} \tag{3.12}$$

Therefore the truncated approximation of the BCH formula presented in the equation (2.12) becomes:

$$
\begin{aligned}
BCH^0(\mathbf{u},\mathbf{v}) &= \mathbf{u}+\mathbf{v}\\
BCH^1(\mathbf{u},\mathbf{v}) &= \mathbf{u}+\mathbf{v}+\frac{1}{2}(J_u\mathbf{v}-J_v\mathbf{u})\\
BCH^{3/2}(\mathbf{u},\mathbf{v}) &= \mathbf{u}+\mathbf{v}+\frac{1}{2}(J_u\mathbf{v}-J_v\mathbf{u})+\frac{1}{12}\big(2J_{\mathbf{u}}J_{\mathbf{u}}\mathbf{v}+2J_{\mathbf{u}}J_{\mathbf{v}}\mathbf{u}-J_{(J_u\mathbf{v}-J_v\mathbf{u})}\mathbf{u}\big)\\
BCH^2(\mathbf{u},\mathbf{v}) &= \mathbf{u}+\mathbf{v}+\frac{1}{2}(J_u\mathbf{v}-J_v\mathbf{u})\\
&\quad+\frac{1}{12}\big(2J_{\mathbf{u}}J_{\mathbf{u}}\mathbf{v}+2J_{\mathbf{u}}J_{\mathbf{v}}\mathbf{u}-J_{(J_{\mathbf{u}}\mathbf{v}-J_{\mathbf{u}}\mathbf{v})}\mathbf{u}+2J_{\mathbf{v}}J_{\mathbf{v}}\mathbf{u}+2J_{\mathbf{v}}J_{\mathbf{u}}\mathbf{v}-J_{(J_u\mathbf{v}-J_v\mathbf{u})}\mathbf{v}\big)
\end{aligned}
$$

Lie brackets of SVF can become extremely small, in particular, as we will see in chapter 5, when the standard deviation of the Gaussian filter that generates the fields is small.
It is not known how to apply Taylor method presented in 2.4.2 for the SVF, but the method based on the parallel transport is a direct consequence of equation (2.14):

$$\mathbf{u}_0 \oplus \mathbf{u}_1 \simeq \mathbf{u}_0 + \exp_e\big(\frac{\mathbf{u}_0}{2}\big)\circ\exp_e(\mathbf{u}_1)\circ\exp_e\big(-\frac{\mathbf{u}_0}{2}\big) - e$$

Here the exponential function can be computed with several algorithms (scaling and squaring, forward Euler, composition method, poser series expansion - see [BZO08] for a comparison of their performances). Following the original setting of the Log-euclidean metric proposed in [ACPA06] we use the scaling and squaring, keeping in mind that this choice affects the results.

# Chapter 4

# Log-composition to Compute the Lie Logarithm

The *logarithm computation problem* can be stated as follows:

*Given $p$ in a Lie group $\mathbb{G}$,*
*what is the element $\mathbf{u}$ in its Lie algebra $\mathfrak{g}$*
*such that* $\exp(\mathbf{u}) = p$ *?*

There are several numerical methods to compute the approximation of the problem's solution. Arsigny, who first pointed the applications of the Lie logarithm in medical image registration in [AFPA06] and [APA06], proposed the Inverse scaling and squaring. In this chapter we investigate other numerical iterative algorithms for the computation of the Lie logarithm, called here *logarithm computation algorithm*. They consists in a modification of the algorithm presented in [BO08a] that is based on the BCH formula, and therefore can be reformulated using the Lie log-composition. It turns out that each of the numerical method to compute the Lie log-composition becomes naturally a numerical method for the logarithm computation. The first step toward this direction is to introduce the space of the approximations of a Lie algebra and a the Lie group.

## 4.1 Spaces of Approximations

As seen in section 3.1 and 3.2, if the element $\mathbf{u}$ of $\mathfrak{se}(2)$ or SVF is small enough we can approximate $\exp(\mathbf{u})$ with $e + \mathbf{u}$. Based on this, we define two approximating functions:

$$\begin{aligned} \text{app} : \mathfrak{g} &\longrightarrow \mathfrak{g}^{\sim} \\ \mathbf{u} &\longmapsto \exp(\mathbf{u}) - e \end{aligned}$$

$$\begin{aligned} \text{App} : \mathbb{G} &\longrightarrow \mathbb{G}^{\sim} \\ \exp(\mathbf{u}) &\longmapsto e + \mathbf{u} \end{aligned}$$

Where $\mathfrak{g}^{\sim}$ is the space of approximations of elements of $\mathfrak{g}$, and $\mathbb{G}^{\sim}$ is the space of approximations of elements in $\mathbb{G}$, defined as

$$\begin{aligned} \mathfrak{g}^{\sim} &:= \{\exp(\mathbf{u}) - e \mid \mathbf{u} \in \mathfrak{g}\} \cup \mathfrak{g} \\ \mathbb{G}^{\sim} &:= \{e + \mathbf{u} \mid \exp(\mathbf{u}) \in \mathbb{G}\} \cup \mathbb{G} \end{aligned}$$

In general $\mathfrak{g}^{\sim} \neq \mathfrak{g}$ and $\mathbb{G}^{\sim} \neq \mathbb{G}$, but in the considered cases of $\mathfrak{se}(2)$ and SVF, when $\mathbf{u}$ is *small enough* it follows that $\exp(\mathbf{u}) - e \in \mathfrak{g}$ and $e + \mathbf{u} \in \mathbb{G}$. Therefore the elements of $\mathfrak{g}^{\sim}$ are compatible with all of the operations of the Lie algebra $\mathfrak{g}$ and the elements of $\mathbb{G}^{\sim}$ are compatible with all of the operations of the Lie group $\mathbb{G}$.
Lets examine what does *small enough* means in these two cases:

$\mathfrak{se}(2)$ - Since $\mathfrak{se}(2)$ and $SE(2)$ are subset of the bigger algebra $SE(2)$ then exp and log can be defined as infinite series. From

$$\exp(\mathbf{u}) = I + \mathbf{u} + O(\mathbf{u}^2)$$

It follows that $\text{app}(\mathbf{u}) - \mathbf{u} \in O(\mathbf{u}^2)$. Thus for all $\mathbf{u}$ in the Lie algebra smaller than $\delta$ for any norm, exists $M(\delta)$ such that

$$||\text{app}(\mathbf{u}) - \mathbf{u}|| < M(\delta)||\mathbf{u}^2||$$

SVF - In case of SVF we do not have any Taylor series and big-O notation available but, according to the proposition 8.6 at page 163 of [You10], if $\mathbf{u}$ is, for any norm, smaller than $\epsilon < 1/C$, where $C$ is the Lipschitz constant in the same norm, then $1 + \mathbf{u}$ is a diffeomorphism. With this condition holds that $\text{SVF}^{\sim} = \text{SVF}$.

Therefore, for each small enough $\mathbf{u}$ in $\mathfrak{se}(2)$ or SVF, and for the definition of the Lie log-composition (equation 2.11) the following properties holds:

1. The approximations $\mathbf{u} \simeq \text{app}(\mathbf{u})$, $\exp(\mathbf{u}) \simeq \text{App}(\exp(\mathbf{u}))$ are bounded.
2. $\mathbf{u} = \mathbf{v} \oplus (-\mathbf{v} \oplus \mathbf{u})$
3. $\text{app}(\mathbf{v} \oplus \mathbf{u}) = \exp(\mathbf{v}) \circ \exp(\mathbf{u}) - 1 \in \mathfrak{g}^{\sim}$

With this machinery, we can reformulate the algorithm presented in [BO08a] for the numerical computation of the Lie logarithm using the Lie log-composition.

## 4.2 The Logarithm Computation Algorithm using Lie Log-composition

If the goal is to find $\mathbf{u}$ when its exponential is known, we can consider the sequence transformations $\{\mathbf{u}_j\}_{j=0}^{\infty}$ that approximate $\mathbf{u}$ as consequence of

$$\mathbf{u} = \mathbf{u}_j \oplus (-\mathbf{u}_j \oplus \mathbf{u}) \Longrightarrow \mathbf{u} \simeq \mathbf{u}_j \oplus \text{app}(-\mathbf{u}_j \oplus \mathbf{u})$$

This suggest that a reasonable approximation for the $(j+1)$-th element of the series can be defined by

$$\mathbf{u}_{j+1} := \mathbf{u}_j \oplus \text{app}(-\mathbf{u}_j \oplus \mathbf{u})$$

If we chose the initial value $\mathbf{u}_0$ to be zero, then the algorithm presented in [BO08a] becomes:

$$\begin{cases} \mathbf{u}_0 = 0 \\ \mathbf{u}_{j+1} = \mathbf{u}_j \oplus \text{app}(-\mathbf{u}_j \oplus \mathbf{u}) \end{cases} \tag{4.1}$$

Making explicit the Lie log-computation and the approximation involved, it follows:

$$\mathbf{u}_{j+1} = \mathbf{u}_j \oplus \big(\exp(-\mathbf{u}_j) \circ \exp(\mathbf{u}) - e\big) \tag{4.2}$$

$$= \log\Big(\exp(\mathbf{u}_j) \circ \exp\big(\exp(-\mathbf{u}_j) \circ \varphi - e\big)\Big) \tag{4.3}$$

where $\exp(\mathbf{u}) = \varphi$ is given by the problem, and $\mathbf{u}_j$ by the previous step. The BCH provides the analytic solution of the second member, while numerical methods to compute the Lie log-composition become numerical methods to compute the Lie logarithm.

### 4.2.1 Truncated BCH Strategy

At each step, we compute the approximation $\mathbf{v}_{j+1}$ with the $k$-th truncation of the BCH formula. The compact form of the algorithm is given by:

$$\begin{cases} \mathbf{u}_0 = 0 \\ \mathbf{u}_{j+1} = \mathrm{BCH}^k(\mathbf{u}_j, \mathrm{app}(-\mathbf{u}_j \oplus \mathbf{u})) \end{cases} \tag{4.4}$$

For $k = 0$, the approximation $\mathbf{u}_{j+1}$ is the sum of the two vectors $\mathbf{u}_j$ and $\mathrm{app}(-\mathbf{u}_j \oplus \mathbf{u})$:

$$\begin{aligned} \mathrm{BCH}^0(\mathbf{u}_j, \mathrm{app}(-\mathbf{u}_j \oplus \mathbf{u})) &= \mathbf{u}_j + \mathrm{app}(-\mathbf{u}_j \oplus \mathbf{u}) \\ &= \mathbf{u}_j + \exp(-\mathbf{u}_j) \circ \varphi - e \end{aligned}$$

When $k = 1$, it results

$$\begin{aligned} \mathrm{BCH}^1(\mathbf{u}_j, \mathrm{app}(-\mathbf{u}_j \oplus \mathbf{u})) &= \mathbf{u}_j + \mathrm{app}(-\mathbf{u}_j \oplus \mathbf{u}) + \frac{1}{2}[\mathbf{u}_j, \mathrm{app}(-\mathbf{u}_j \oplus \mathbf{u})] \\ &= \mathbf{u}_j + \exp(-\mathbf{u}_j) \circ \varphi - e + \\ &+ \frac{1}{2}(\mathbf{u}_j \cdot (\exp(-\mathbf{u}_j) \circ \varphi - e) - (\exp(-\mathbf{u}_j) \circ \varphi - e) \cdot \mathbf{u}_j) \end{aligned}$$

Where $\circ$ the product is the product using the Jacobian (see equation (3.12)), for SVF, or the matrix product.

For $k = 2$ it become

$$\begin{aligned} \mathrm{BCH}^2(\mathbf{u}_j, \mathrm{app}(-\mathbf{u}_j \oplus \mathbf{u})) &= \mathbf{u}_j + \mathrm{app}(-\mathbf{u}_j \oplus \mathbf{u}) + \frac{1}{2}[\mathbf{u}_j, \mathrm{app}(-\mathbf{u}_j \oplus \mathbf{u})] + \\ &+ \frac{1}{12}\Big([\mathbf{u}_j, [\mathbf{u}_j, \mathrm{app}(-\mathbf{u}_j \oplus \mathbf{u})]] + \\ &+ [\mathrm{app}(-\mathbf{u}_j \oplus \mathbf{u}), [\mathrm{app}(-\mathbf{u}_j \oplus \mathbf{u}), \mathbf{u}_j]]\Big) \\ &= \mathbf{u}_j + \exp(-\mathbf{u}_j) \circ \varphi - e + \frac{1}{2}[\mathbf{u}_j, \exp(-\mathbf{u}_j) \circ \varphi - e] + \\ &+ \frac{1}{12}\Big([\mathbf{u}_j, [\mathbf{u}_j, \exp(-\mathbf{u}_j) \circ \varphi - e]] + \\ &+ [\exp(-\mathbf{u}_j) \circ \varphi - e, [\exp(-\mathbf{u}_j) \circ \varphi - e, \mathbf{u}_j]]\Big) \end{aligned}$$

When considering $k = \infty$ and so the analytic solution of the Lie log-composition provided by the BCH formula, the following theorem, presented in [BO08a], provides an error bound:

**Theorem 4.2.1** (Bossa)**.** The iterative algorithm

$$\begin{cases} \mathbf{u}_0 = 0 \\ \mathbf{u}_{j+1} = \mathbf{u}_j \oplus \mathrm{app}(-\mathbf{u}_j \oplus \mathbf{u}) \end{cases} \tag{4.5}$$

converges to $\mathbf{v}$ with error $\delta_n \in \mathbb{G}$, where

$$\delta_n := \log(\exp(\mathbf{v}) \circ \exp(-\mathbf{v}_n)) \in O(||p - e||^{2^n})$$

We observe that this upper limit can be computed only when a closed-form for the Lie log-composition is available, as for example $\mathfrak{se}(2)$.

### 4.2.2 Parallel Transport Strategy

If we apply the parallel transport method for the computation of the Lie log-composition, we obtain another version of the algorithm:

$$\begin{cases} \mathbf{u}_0 = \mathbf{0} \\ \mathbf{u}_{j+1} = \mathbf{u}_j + \exp(\frac{\mathbf{u}_j}{2}) \circ \exp\Big(\text{app}(-\mathbf{u}_j \oplus \mathbf{u})\Big) \circ \exp(-\frac{\mathbf{u}_j}{2}) - e \end{cases} \tag{4.6}$$

That is computed as:

$$\mathbf{u}_{j+1} = \mathbf{u}_j + \exp(\frac{\mathbf{u}_j}{2}) \circ \exp\Big(\exp(-\mathbf{u}_j) \circ \varphi - e\Big) \circ \exp(-\frac{\mathbf{u}_j}{2}) - e$$

We notice that mixing the operation of composition, sum and scalar product makes sense when the involved vectors are *small enough*, as stated in 4.1. Analytical computation of an upper bound error is not straightforward in this case.

### 4.2.3 Symmetrization Strategy

The algorithm 4.1 could have been reformulated alternatively as $\mathbf{u}_{j+1} = \text{app}(\mathbf{u} \oplus -\mathbf{u}_j) \oplus \mathbf{u}_j$. The Lie log-composition is not symmetric therefore the two version in some cases may not return the same value. We consider

$$\begin{cases} \mathbf{u}_0 = 0 \\ \mathbf{u}_{j+1} = \mathbf{u}_j \oplus \frac{1}{2}\big(\text{app}(-\mathbf{u}_j \oplus \mathbf{u}) + \text{app}(\mathbf{u} \oplus -\mathbf{u}_j)\big) \end{cases} \tag{4.7}$$

that using the BCH approximation of degree 1, becomes:

$$\begin{cases} \mathbf{u}_0 = 0 \\ \mathbf{u}_{j+1} = \mathbf{u}_j + \frac{1}{2}\big(\exp(-\mathbf{u}_j) \circ \varphi) - e + \varphi \circ \exp(-\mathbf{u}_j) - e\big) \end{cases} \tag{4.8}$$

Experimental results of the methods presented in this section are show in section 5.4.

# Chapter 5

# Experimental Results

Aim of this chapter is to show most relevant results of the numerical methods investigated for the computation of the log-composition $\mathbf{u}_0 \oplus \mathbf{u}_1 = \log(\exp(\mathbf{u}_0) \circ \exp(\mathbf{u}_1))$.

We use both synthetic data, produced with randomization, and real data, collected from the ADNI (Alzheimer Disease Neuroimaging Initiative) [JBF+08]. Computations are performed with a software written in Python (repository available on the UCL CMIC gitlab https://cmiclab.cs.ucl.ac.uk), based on the following libraries: numpy, matplotlib [Hun07], math, scipy [JOP+ ], nibabel, timeit, random, as well as on the library NiftyBit, implemented by Pancaj Daga. In addition the software NiftyReg [MRT+10] has been used to obtain SVF from patients scans.

## 5.1 Log-composition for $\mathfrak{se}(2)$

There are several norms in the space of $3 \times 3$ squared matrices that can be used to measure computations results in $\mathfrak{se}(2)$. For our tests we chose the Frobenius norm:

$$||(\theta, dt^x, dt^y)||_{\text{fro}} = \sqrt{2\theta^2 + (dt^x)^2 + (dt^y)^2} \qquad (\theta, dt^x, dt^y) \in \mathfrak{se}(2)$$

Numerical measurement have shown that for the studied cases, no qualitative differences are detected if $l^2$ norm is chosen instead.

### 5.1.1 Methods and Results

To compare the errors of the computation of the log-composition for the methods here presented, two sets of 3000 transformations of elements in $\mathfrak{se}(2)$ are randomly sampled with increasing norms in the interval $[0.1, 2.0]$. Since the precision of the results is affected by the norm of the involved SVF, this interval is divided into 6 segments delimited by the values $I = [0.1, 0.42, 0.73, 1.05, 1.73, 1.68]$; choosing a couple of subintervals $[I(n_0), I(n_0 + 1)]$ and $[I(n_1), I(n_1 + 1)]$, two sets of 500 transformations $\{dr_0^{(j)}\}_{j=1}^{500}$, $\{dr_1^{(j)}\}_{j=1}^{500}$ are sampled with norms belonging to the selected intervals:

$$\begin{aligned} j = 1, ..., 500 \qquad & n_0, n_1 = 0, ..., 5 \\ ||dr_0^{(j)}||_{\text{fro}} & \in [I(n_0), I(n_0 + 1)] \\ ||dr_1^{(j)}||_{\text{fro}} & \in [I(n_1), I(n_1 + 1)] \end{aligned}$$

If $M$ is one of the numerical methods presented in section 3.1 for the computation of the log-composition - $\text{BCH}^0, \text{BCH}^1, \text{BCH}^2$, Taylor, p.t. (parallel transport) - then the error between

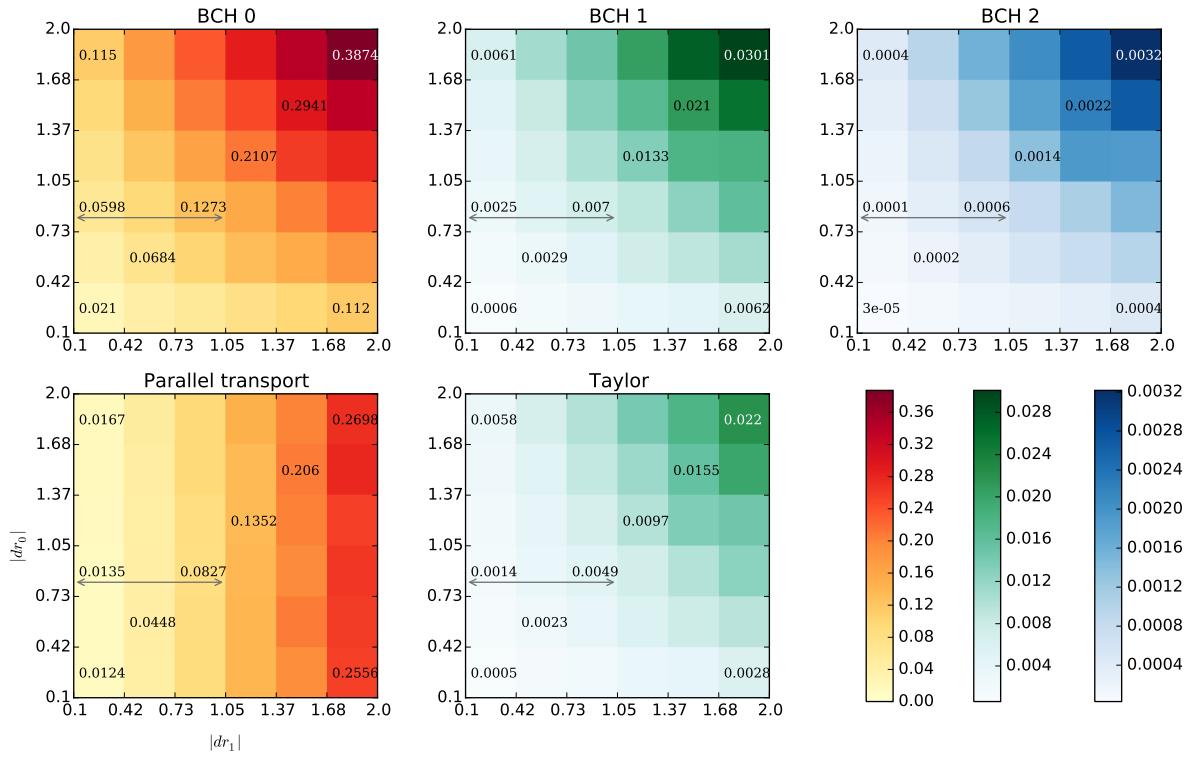



Figure 5.1: Comparisons of the errors for each numerical method to compute the Log-composition $dr_0 \oplus dr_1$ in $\mathfrak{se}(2)$. Truncated BCH of degrees 0,1,2, parallel transport method and Taylor method are considered for different values of the norm of $dr_1$ (x-axes) and norm of $dr_0$ (y-axes). The value of each square corresponds to the average error of 500 random samples in each of the 6 sub-intervals between 0.1 and 2.0. As expected errors increases with the norm for all of the methods. Errors with BCH 0 and parallel transport method are comparable, but the parallel transport method is not symmetric and has better performance when $dr_1$ is small. BCH[1] and Taylor are comparable as well, and they are both symmetric, but the best performance is provided by the BCH 2. Details of the value under the *gray arrows* are shown in the box-plot 5.1 where means, variance, range, quartiles and outliers are visualized.

the ground truth and the approximation provided by one of these numerical methods is given by

$$\text{Error}(dr_0, dr_1, M) := ||dr_0 \oplus dr_1 - M(dr_0, dr_1)||_{\text{fro}}$$

In figure 5.1, each of the figure corresponds to a different method and each of the grade scale is the value computed with the function:

$$f(n_0, n_1, M) = \mathbb{E}\Big(\{\text{Error}(dr_0^{(j)}, dr_1^{(j)}, M)\}_{j=1}^{500}\Big)$$

where the norm of $dr_0^{(j)}$ belongs to the interval $[I(n_0), I(n_0+1)]$ and the norm of $dr_1^{(j)}$ belongs to $[I(n_1), I(n_1+1)]$, and where $\mathbb{E}$ is the mean value.
The details for a chosen interval, indicated by the gray arrows in each plot, can be visualized in the box-plot 5.2.

The method based on the $BCH^0$ provides the worst results. It does not involves any Lie bracket and so it does not take into account the geometrical curvature of the Lie group.

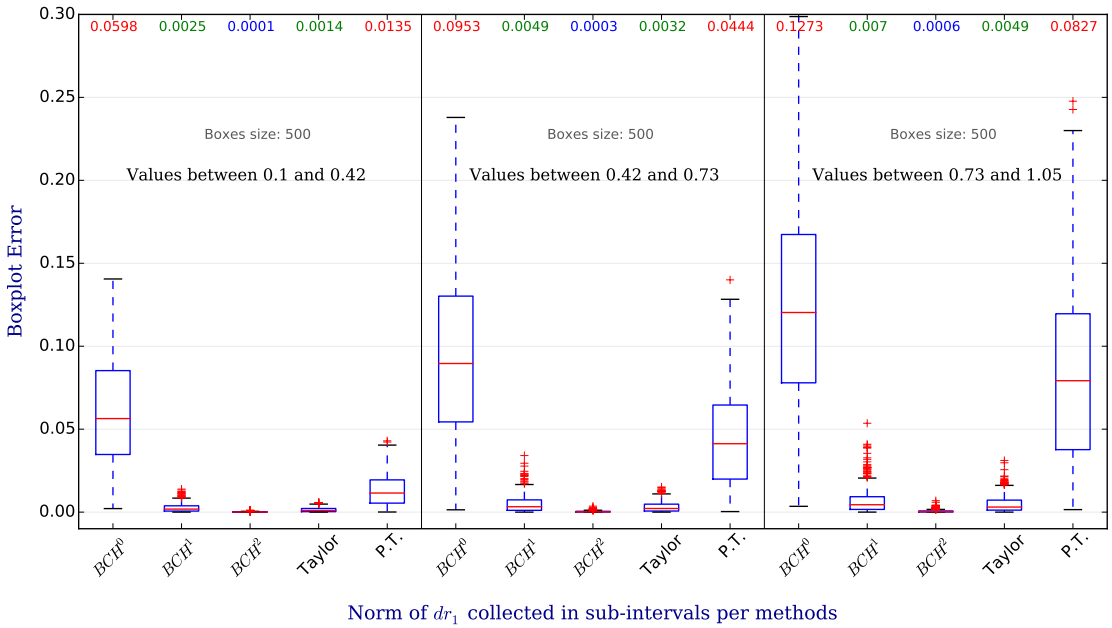



Figure 5.2: Errors of the numerical methods for the computation of the Lie log-composition of $dr_0 \oplus dr_1$ in $\mathfrak{se}(2)$. Norm of $dr_0$ is in the interval $[0.37, 1.05]$, norm of $dr_1$ in the interval $[0.1, 1.05]$ divided in 3 segments: $[0.1, 0.42]$, $[0.42, 0.73]$, $[0.73, 1.05]$. Mean values of each box are shown in the first row in different colors. Shown data corresponds to a section of the image scale 5.1, indicated by a gray arrow. As expected all of the error means increase with the of norm of $dr_1$, but the rate of growth is different for each method.

Parallel transport method tries to compensate the curvature using a geometrical approach. As expected from the formula, and as it can be seen clearly in the image scale, it is not a symmetric method. It provides better results than the $BCH^0$; for small norm of $dr_1$, results are close to the one obtained with $BCH^1$ when norms of $dr_0$ and $dr_1$ are below 1.3.

Lie log-composition based on Taylor method has slightly better results than the $BCH^1$, but do not reach $BCH^2$, which provides the best results. This may be due to the fact that the Taylor method belongs to $\mathcal{O}(dr_1^2)$ while the $BCH^2$ involves the Lie bracket $[dr_0, [dr_0, dr_1]] + [dr_1, [dr_1, dr_0]]$. Even if the truncated $BCH$ does not have a known asymptotic error (or big-O notation), we can see experimentally that $BCH^2$ have a larger asymptotic order of converges than $\mathcal{O}(dr_1^2)$, in $\mathfrak{se}(2)$.

## 5.2 Log-composition for SVF

Before getting into the results for the Lie log-composition of SVF we show how random SVF are created and, in particular, how the norm of the approximation of $\mathbf{u}_0 \oplus \mathbf{u}_1$ is compared with the ground truth when this is not directly available.

### 5.2.1 Methods: random generated SVF

A random generated SVF is a 5-dimensional matrix with the structure presented in the formula (3.11). The vectors at each point of the grid are generated in two phases. In the first one, the values of the coordinates of the vectors are randomly sampled from a normal

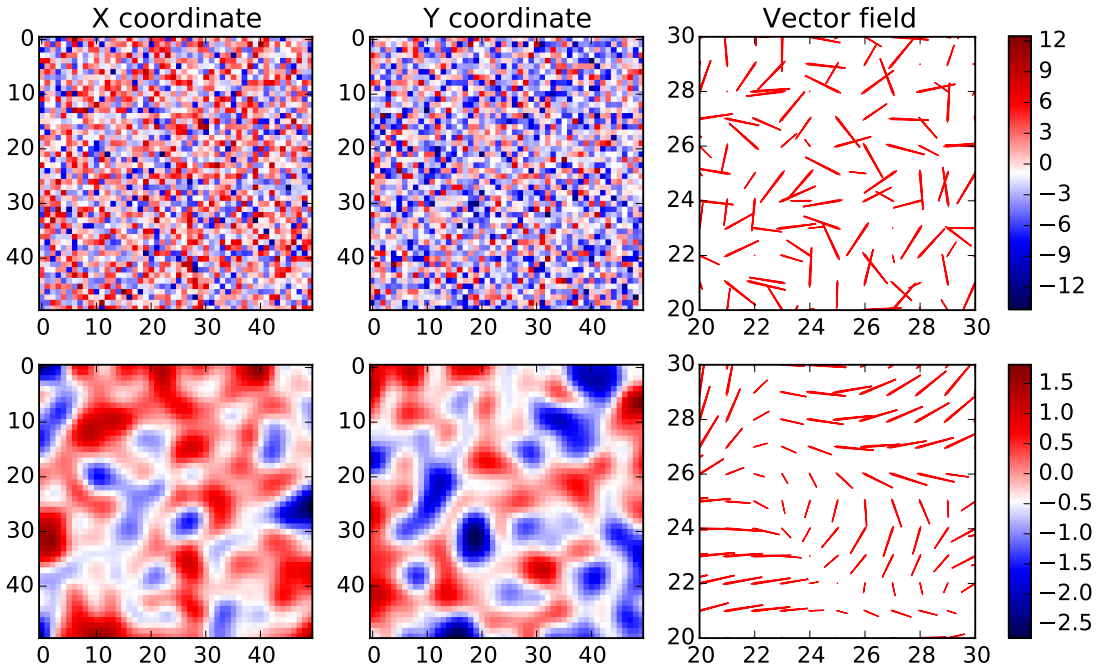



Figure 5.3: Random generated vector field before and after the Gaussian smoother. In the first row is shown a random generated vector field of dimension $50 \times 50 \times 2$. Values of the vectors at each pixel are sampled from a random variable with normal distribution of mean 0 and sigma 4. The second row shows the same random vector field after a Gaussian smoothing of sigma 2 (the code is based on the scipy library ndimage.filters.gaussian_filter). The last column shows the quiver of the vector field in the squared subregion of size $10 \times 10$ at the point $(20, 20)$. From the colorscale it is also possible to see that the values distribution of the filtered image have lost their symmetry.

distribution of mean zero and standard deviation $\sigma_{\text{init}}$. In the second phase vectors are smoothed with a Gaussian filter with standard deviation $\sigma_{\text{gf}}$. In figure 5.3 it is possible to see the effects of the two phases on a bi-dimensional $50 \times 50$ image.

After discretization, the norm of an SVF can be computed from the discretized version of the metric presented in equation (3.8). The norm $l^2$ is considered instead of $L^2$ and $\Delta\Omega$, discretization of the domain, substitutes $\Omega$:

$$||\mathbf{u}|| = \Big( \sum_{\mathbf{x} \in \Delta\Omega} ||\mathbf{u}(\mathbf{x})||^2_{l^2} \Big)^{1/2}$$

This norm coincides with the Frobenius Norm of the 5-dimensional matrices $\mathbf{u}$. When an SVF $\mathbf{u}$ is exponentiated in the Lie algebra, $\exp(\mathbf{u}) = \varphi$, we can rely on the fact that $\varphi - 1$ is a vector field whose norm can be computed with the discretization of the metric presented in equation (3.9):

$$||\varphi|| = \Big( \sum_{\mathbf{x} \in \Delta\Omega} ||\mathcal{V}(\varphi)(\mathbf{x})||^2_{l^2} \Big)^{1/2} = \Big( \sum_{\mathbf{x} \in \Delta\Omega} ||\varphi(\mathbf{x}) - \mathbf{x}||^2_{l^2} \Big)^{1/2} \tag{5.1}$$

To distinguish the norm of the SVF from the norm of the vector field associated to its exponential, we call the latter DS-norm (displacement norm); as before, it coincides with the Frobenius norm of the 5-dimensional matrices $\varphi - I$.

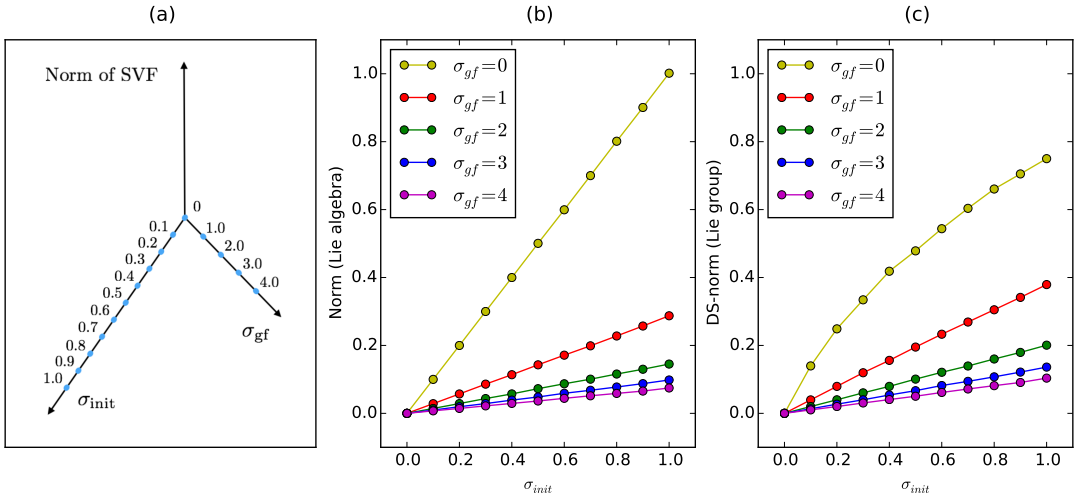



Figure 5.4: Relationships between the initial standard deviation $\sigma_{\text{init}}$ that defines the random SVF (stationary velocity field), the standard deviation of the Gaussian filter $\sigma_{\text{gf}}$ utilized to regularize the SVF and its norm. Figure (a) represents schematically the two factors that define the norm of an SVF and with the blue dots we emphasized the values that has been chosen for the numerical computations proposed in (b) and (c). Figure (b) shows the mean of the norm of 10 random generated SVF, with initial standard deviation $\sigma_{\text{init}}$ (on the x-axis) and Gaussian filter with standard deviation $\sigma_{\text{gf}}$ (in different colors). Figure (c) shows the norm of the same element after exponentiating and so after having them in the Lie algebra. It is important to remark that it is not possible in general define a norm on a group. Nevertheless for matrices and for SVF it is possible to extend the norm from the Lie algebra to the Lie group, as proposed in chapter 3 with the definition of displacement field norm (DS-norm) proposed in equation (5.1).

We can see how these two norms are related in figure 5.4. The initial standard deviations $\sigma_{\text{init}}$ that defines the SVF are related with their norm before and after the exponentiation for 5 different choices of the value of the standard deviation of the Gaussian filter $\sigma_{\text{gf}}$. For the extreme case $\sigma_{\text{gf}} = 0$, where the SVF is not properly defined, the norm after the exponentiation does not maintain linearity, but in all the other cases an element in the Lie algebra $\mathbf{u}$ increases the norm after the exponentiation. Moreover, for a fixed $\sigma_{\text{gf}}$ in both algebraic structures the norms shows a linear trend with the increase of $\sigma_{\text{init}}$. An increase in $\sigma_{\text{gf}}$ implies a decrease in the slope of the linear model.

The linear regression of the model for each $\sigma_{\text{gf}}$ are given, in Cartesian coordinate by:

$$y = m_{\text{alg}}(\sigma_{\text{gf}})x \qquad \sigma_{\text{gf}} \geq 0 \qquad\qquad y = m_{\text{grp}}(\sigma_{\text{alg}})x \qquad \sigma_{\text{alg}} \geq 0 \tag{5.2}$$

Where we indicated with $m_{\text{alg}}$ and $m_{\text{grp}}$ angular coefficients for the results obtained in the Lie algebra and in the Lie group respectively (figure 5.4 (b) and (c)). They follow an exponential model, given by

$$m_{\text{alg}}(\sigma_{\text{gf}}) = \alpha_0 e^{-\beta_0 \sigma_{\text{gf}}} + \gamma_0 \qquad m_{\text{grp}}(\sigma_{\text{gf}}) = \alpha_1 e^{-\beta_1 \sigma_{\text{gf}}} + \gamma_1 \tag{5.3}$$

Where the parameters $\alpha_i, \beta_i, \gamma_i$ for $i = 1, 2$ can be computed numerically using an exponential regression algorithm:

| $\alpha_0$ | $\beta_0$ | $\gamma_0$ | $\alpha_1$ | $\beta_1$ | $\gamma_1$ |
|---|---|---|---|---|---|
| 0.91422836 | 1.48548466 | 0.08393943 | 0.67302265 | 0.82680977 | 0.07765811 |

These values are useful when, given two values among $\sigma_{\text{init}}$, $\sigma_{\text{gf}}$ and the $||\mathbf{u}||$, we want to know the third one.

After showing how a random SVF is generated by the parameters $\sigma_{\text{init}}$ and $\sigma_{\text{gf}}$, and what is the relationship between the parameters and the resulting norm in both Lie algebra and Lie group, we can move toward the results of the numerical method for the Lie log-composition obtained with these objects.

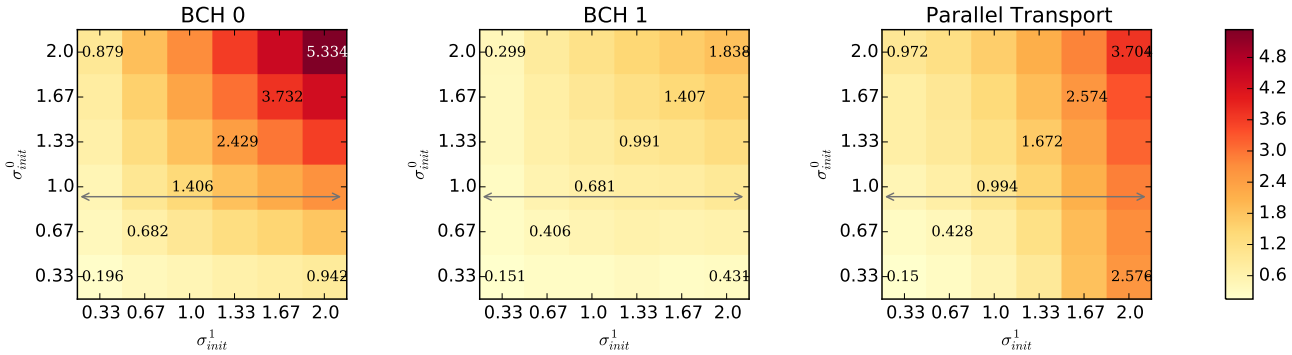



Figure 5.5: Mean errors for the numerical computation of the Log-composition of randomly generated stationary velocity fields (SVF). Initial standard deviations of the SVF $\sigma^0_{\text{init}}$ and $\sigma^1_{\text{init}}$ are given by the values on the axis for each sampling of 15 elements. The error is computed using the formula (5.5). When the norm of $\mathbf{u}_1$ is small (see figure 5.4 to infer the norm from the standard deviations), parallel transport method and truncated BCH of degree 1 have comparable results, but parallel transport, as expected from the formula, is not symmetric respect to the size of the input vectors. Results of another sampling with the value of $\sigma^0_{\text{init}}$ and $\sigma^1_{\text{init}}$ are shown in figure 5.6. The reason why results obtained $\text{BCH}^{1.5}$ and $\text{BCH}^2$ is presented in figure 5.8.

### 5.2.2 Lie Log-composition for synthetic SVF

This section shows the results of the numerical computation of the Lie log-composition $\mathbf{u}_0 \oplus \mathbf{u}_1 = \log(\exp(\mathbf{u}_0) \circ \exp(\mathbf{u}_1))$ when $\mathbf{u}_0$ and $\mathbf{u}_1$ are synthetic SVF. Despite the lack of a ground truth for the SVF it is reasonable to compare the numerical approximation of the exponential of $\mathbf{u}_0 \oplus \mathbf{u}_1$ with $\exp(\mathbf{u}_0) \circ \exp(\mathbf{u}_1)$. The norm utilized is the one proposed in the equation 93.10):

$$\text{Error}_{\oplus}(\mathbf{u}_0, \mathbf{u}_1) = \Big( \int_{\Omega} ||\mathcal{V}(\exp(\mathbf{u}_0 \oplus \mathbf{u}_1)) - \mathcal{V}(\exp(\mathbf{u}_0) \circ \exp(\mathbf{u}_1))||^2_{L^2} d\mathbf{x} \Big)^{1/2} \tag{5.4}$$

that, when discretized becomes

$$\text{Error}_{\oplus}(\mathbf{u}_0, \mathbf{u}_1) = \Big( \sum_{\mathbf{x} \in \Delta\Omega} ||\exp(\mathbf{u}_0 \oplus \mathbf{u}_1)(\mathbf{x}) - \big(\exp(\mathbf{u}_0) \circ \exp(\mathbf{u}_1)\big)(\mathbf{x})||^2_{l^2} \Big)^{1/2} \tag{5.5}$$

For the unknown analytical result of $\mathbf{u}_0 \oplus \mathbf{u}_1$ the error of the above equation is 0, since $\exp(\mathbf{u}_0 \oplus \mathbf{u}_1) = \exp(\mathbf{u}_0) \circ \exp(\mathbf{u}_1)$. When we use one of the introduced numerical method we obtain the results presented in figure 5.5 and 5.6.

The limitation of this strategy to compute the error of the numerical method without a ground truth is that it is based on the numerical algorithm utilized for the computation of the Lie exponential (the scaling and squaring in this experiment [ACPA06]). But this limitation tend to not bias the results, since scaling and squaring is utilized it the both sides of the difference in the computation of the error.

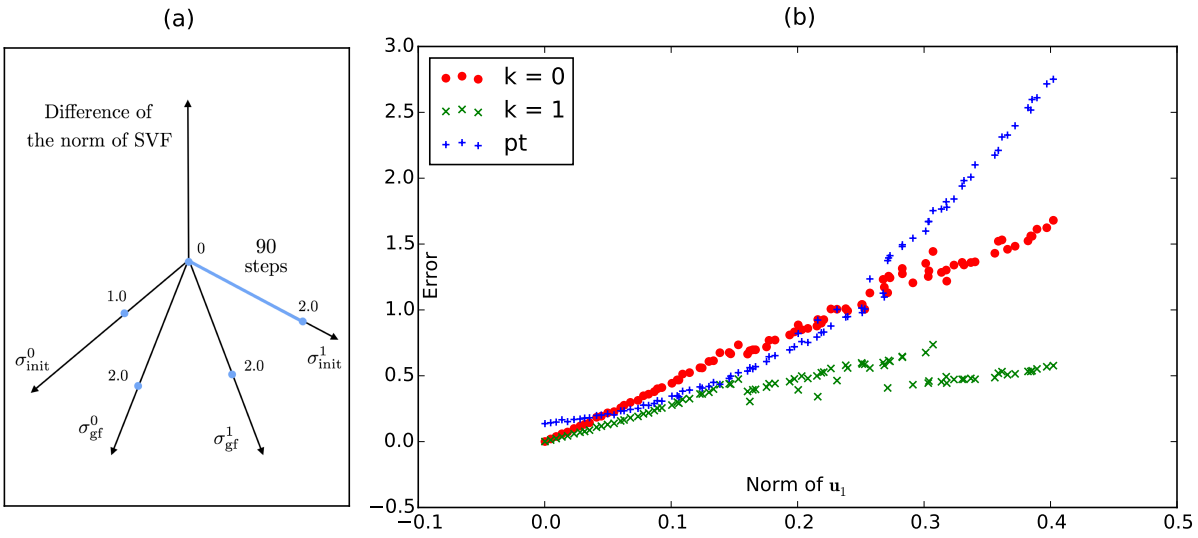



Figure 5.6: Comparisons of the errors of numerical computations of the Lie log-composition $\mathbf{u}_0 \oplus \mathbf{u}_1$ with the method of truncated BCH of degree 0,1 and parallel transport. Parameters' values of the random generated SVF are schematically represented in figure (a). A set of 90 SVF $\mathbf{u}_0$ are generated with fixed parameters $\sigma^0_{\text{gf}} = 2.0$ and $\sigma^0_{\text{init}} = 1.0$; a second set of 90 SVF $\mathbf{u}_1$, are generated with the parameters $\sigma^1_{\text{gf}} = 2.0$ and $\sigma^1_{\text{init}}$ uniformly scattered in the the interval $(0.0, 2.0)$. On the x-axis fo figure (b) is shown the value of the resulting norm of $\mathbf{u}_1$ for the chosen parameters, while on the y-axes are shown the values of the error for the numerical computation of the Log-composition $\mathbf{u}_0 \oplus \mathbf{u}_1$ for each of the chosen methods.

In figure 5.5, we can see the differences between the errors obtained with the numerical methods based on $\text{BCH}^0$, $\text{BCH}^1$ and parallel transport. To each square corresponds the mean of 15 log-compositions between the SVF $\mathbf{u}_0$ and $\mathbf{u}_1$ randomly generated. The parameters $\sigma^0_{\text{init}}$ and $\sigma^1_{\text{init}}$ are equals to one of the value in the array $(0.33, 0.67, 1, 1.33, 1.37, 2.0)$ while the standard deviation of the Gaussian filter $\sigma^0_{\text{gf}}$ and $\sigma^1_{\text{gf}}$ are constant and equal to 2.0. As previously noticed for matrices, the methods based on the truncated BCH are symmetric while the same does not happen for the parallel transport.

The SVF in the subinterval indicated by the gray arrow on the figure 5.5 are shown in figure 5.6, where the values of $\mathbf{u}_1$ are equally distributed. From there we can see that parallel transport works better than the $\text{BCH}^0$ only when the norm of $\mathbf{u}_1$ is small enough.

In the results just shown there is a notable absent: the numerical computation of the Lie log-composition based on truncated BCH of order greater than 1. The next section explains why, and in particular why we are interested in BCH-free numerical methods for the log-composition, as the one based on parallel transport.

### 5.2.3 Truncated BCH formula: The problem of the Jacobian matrix.

As shown in figure 5.1, for the finite dimensional case, the truncated BCH of degree 2 provides the best results over the Taylor method and the parallel transport method.

We would expect something similar for SVF, but in this case the use of the truncated BCH of degree greater than 1 is problematic, because Jacobian matrices are involved.

On one side, every time the differentiation of a vector is required, results are unstable and sensitive to noise, because of the finite difference method utilized for the numerical

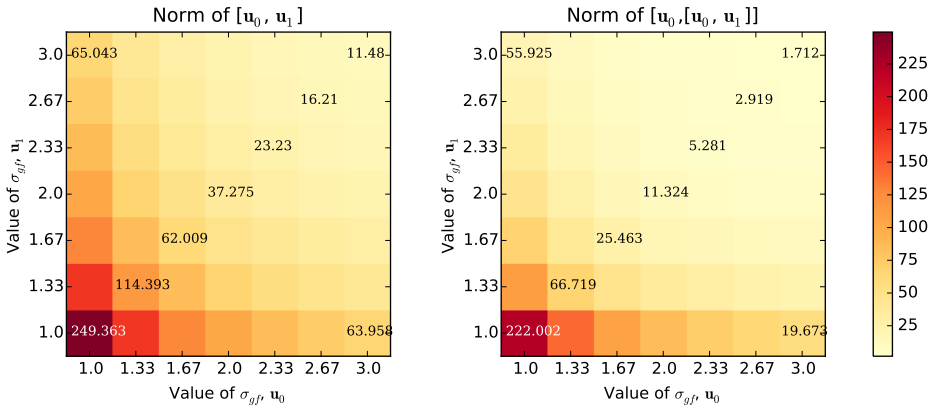



Figure 5.7: Relationships between the standard deviation of the Gaussian smoother that generates the SVF and the norm of the Lie bracket. Each square contains the means of 10 Lie bracket (left) or nested Lie bracket (right) generated with initial standard deviation equals to 2 and standard deviation of the gaussian smoother $\sigma_{\text{gf}}$ indicated on the axes.

approximation. On the other side, in figure 5.7 we can see that the smoother are the SVF the smaller is the norm of resulting Lie brackets. In consequence of this, for a couple of "very smooth" stationary velocity field, the higher term of the BCH carry small information extremely sensitive to noise.

The boxplot in figure 5.8 shows that an increase in the degree of the truncated BCH does not necessarily imply better results, in particular when the involved SVF have been generated with a small Gaussian smoother.

## 5.3 A Problem for Three Brains

The experiments performed on synthetic data provide important informations to validate and compare the methods, but do not give informations about what may happen for real cases.

To validate the method for real cases, one of the possibility is to embed the method in a diffeomorphic demon registration algorithm and compare the results obtained with different numerical methods of the Lie log-composition to compute the update. Since computation of the update is only a small component of the registration algorithm, using this strategy, it may be difficult to study the impact of the Lie log-composition over other components that play a more important role in the registration algorithm (as the optimization strategy).

For this reason we chose to validate the numerical methods for the log-composition using the same strategy utilized for synthetic data, using SVF from longitudinal patient scans.

### 5.3.1 Design of Experiment

We design a simple experiment that involves three T1 MRI longitudinal scans of the same subject in three points in time $A$, $B$ and $C$. $A$ is the baseline, $B$ is the first follow-up, and $C$ is the second follow-up.

Let $\varphi_A^B$ be the non-rigid transformation from $A$ to $B$ that aligns the image $B$ with $A$, $\varphi_B^C$ from $B$ to $C$ and $\mathbf{u}_A^B$, $\mathbf{u}_B^C$ the corresponding SVF. Than for the analytic solution of the Lie log-composition, it follows that

$$\exp(\alpha_0 \mathbf{u}_B^C \oplus \alpha_1 \mathbf{u}_A^B) = \exp(\alpha_0 \mathbf{u}_B^C) \circ \exp(\alpha_1 \mathbf{u}_A^B)$$

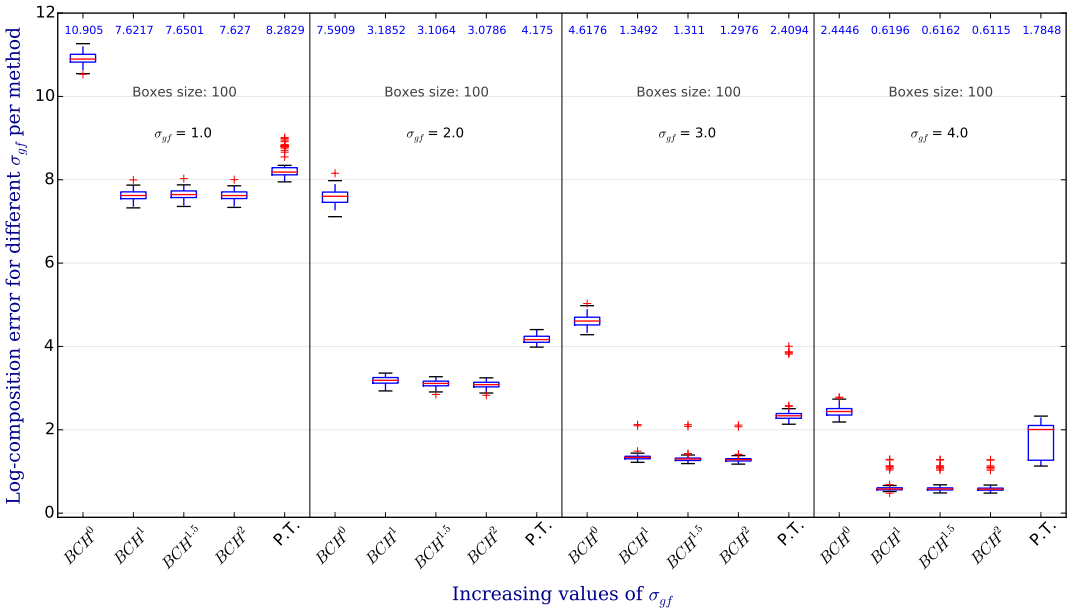



Figure 5.8: Boxplot to compare the error between truncated BCH methods of degree 0, 1, 1.5 and 2. The size of each box is 100 and the approximation of $\mathbf{u}_0 \oplus \mathbf{u}_1$ is performed with $||\mathbf{u}_0|| = 1.0$ and $||\mathbf{u}_0|| = 0.1$. The standard deviation of the Gaussian filter $\sigma_{\text{gf}}$ belongs to the set $(1.0, 2.0, 3.0, 4.0)$ and the initial standard deviation is computed such that $\sigma_{\text{init}} = ||\mathbf{u}||/m_{\text{alg}}(\sigma_{\text{gf}})$ according to the formula (5.2). With this strategy we have been able to compare vector of constant norm generated with increasing values for $\sigma_{\text{gf}}$. The numbers written in blue above each boxes represents the mean value of the errors. For small $\sigma_{\text{gf}}$, an increase in the order of the approximation does not always corresponds to a decrease in the error, and in general there no great improvements can be registered when the degree is greater than 1.

where $\alpha_0$ and $\alpha_1$ are two real numbers between 0 and 1 that are used to reduce the length of the vectors of the SVF. We can use again the equation 5.5, to compute the error of the numerical methods:

$$E^M(\mathbf{u}_A^B, \mathbf{u}_B^C, p_0, p_1) = \\ = \Big( \sum_{\mathbf{x} \in \Delta\Omega} ||\exp(M(p_0 \mathbf{u}_B^C \oplus p_1 \mathbf{u}_A^B))(\mathbf{x}) - \big(\exp(p_0 \mathbf{u}_B^C) \circ \exp(p_1 \mathbf{u}_A^B)\big)(\mathbf{x})||_{l^2}^2 \Big)^{1/2} \quad (5.6)$$

After the computation, error is then normalized by the square root of the product of the dimension of the $SVF$.

### 5.3.2 Results

The dataset has been collected from ADNI (Alzheimer Disease Neuroimaging Initiative) [JBF+08] and consists of 3 longitudinal scans of 16 control subjects (CTL). The baseline $A$ at time 0 is followed by a first follow-up $B$, after three months, and a second follow up $C$, after 6 months. The SVF of the transformations are obtained using NiftyReg [MRT+10].

Figure 5.10 shows the results of the normalized equation (5.6) where $M$ is one of the numerical methods based on $\text{BCH}^0$, $\text{BCH}^1$ and parallel transport (P.T.). Axes indicates

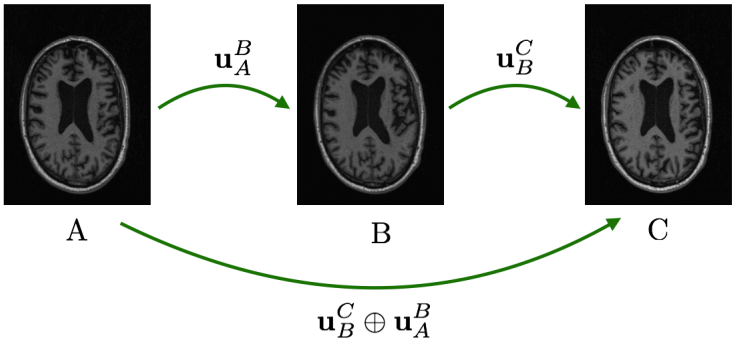



Figure 5.9: Design of the experiment with SVF obtained with real data. Three figures represents three longitudinal scans of the same patient at time $A$, $B$ and $C$. The vector $\mathbf{u}_A^B$ is the SVF that corresponds to the non-rigid transformation from $A$ to $B$ that aligns the image $B$ with $A$. Equivalently $\mathbf{u}_B^C$ is the SVF from $B$ to $C$, and $\mathbf{u}_A^C$ from $A$ to $C$. With the lack of a ground truth to compute the Lie log-composition $\mathbf{u}_B^C \oplus \mathbf{u}_A^B = \log(\exp(\mathbf{u}_B^C) \circ \exp(\mathbf{u}_A^B))$, we compared the composition of the exponential of the SVF with the exponentiated log-composition. This led to the formula (5.6) for the computation of the errors, whose normalized results are shown in figure 5.10.

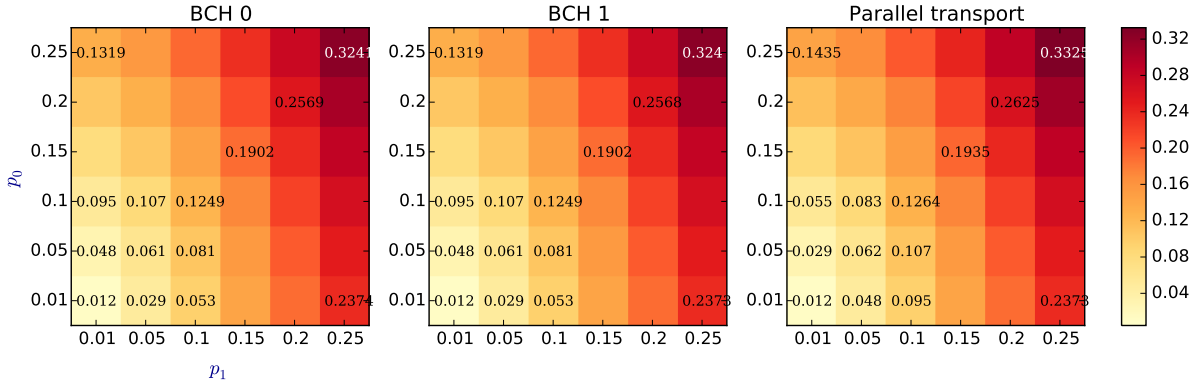



Figure 5.10: Lie Log-composition errors for a dataset of 16 MRI T1 brain images scan of control subjects. Three longitudinal scans $A$, $B$ and $C$ are available for each subject, and corresponds to a scan at time 0, after three months and after 6 month from the 0. The design of the experiment is described in figure 5.9, and the error of the Lie log-composition is computed with the formula (5.6).

the values for $p_1$ and $p_0$, that corresponds to the following average norms of the 16 SVF considered:

| 0.01 | 0.05 | 0.10 | 0.15 | 0.20 | 0.25 |
|---|---|---|---|---|---|
| 0.0045 | 0.0227 | 0.0454 | 0.0682 | 0.0909 | 0.1136 |

Results are consistent with the one obtained for synthetic data except when the norm of $\mathbf{u}_A^B$ si below or equal 0.0045. In this case parallel transport method provides slightly better results than the BCH[1].

A statistical test of significance shows that differences between results obtained with Parallel transport and BCH[0] are not relevant. The corresponding p-value of the hypothesis

that both of the samplings have the same unknown mean showed in the following table are all closed to 1:

| | | | | | | |
|---|---|---|---|---|---|---|
| 0.25 | 0.99983489 | 0.99311505 | 0.97979104 | 0.96440304 | 0.9476505 | 0.9307285 |
| 0.20 | 0.9999575 | 0.99648483 | 0.98591045 | 0.97172936 | 0.95545293 | 0.93867622 |
| 0.15 | 0.99986455 | 0.99790022 | 0.99036937 | 0.9788054 | 0.96428303 | 0.94848053 |
| 0.10 | 0.99969772 | 0.99865145 | 0.99284918 | 0.98343123 | 0.97087921 | 0.9565912 |
| 0.05 | 0.9997416 | 0.99892752 | 0.99417358 | 0.98629071 | 0.97543413 | 0.96264026 |
| 0.01 | 0.99961913 | 0.99927621 | 0.99524053 | 0.98847805 | 0.9789583 | 0.96750909 |
| | 0.01 | 0.05 | 0.10 | 0.15 | 0.20 | 0.25 |

Looking at the picture, the differences of the mean between $\mathrm{BCH}^0$ and $\mathrm{BCH}^1$ can be seen only in some cases after the third decimal. Results obtained with $\mathrm{BCH}^{1.5}$ and $\mathrm{BCH}^2$ are not even showed because they do not lead to any improvement, in particular for small SVF.

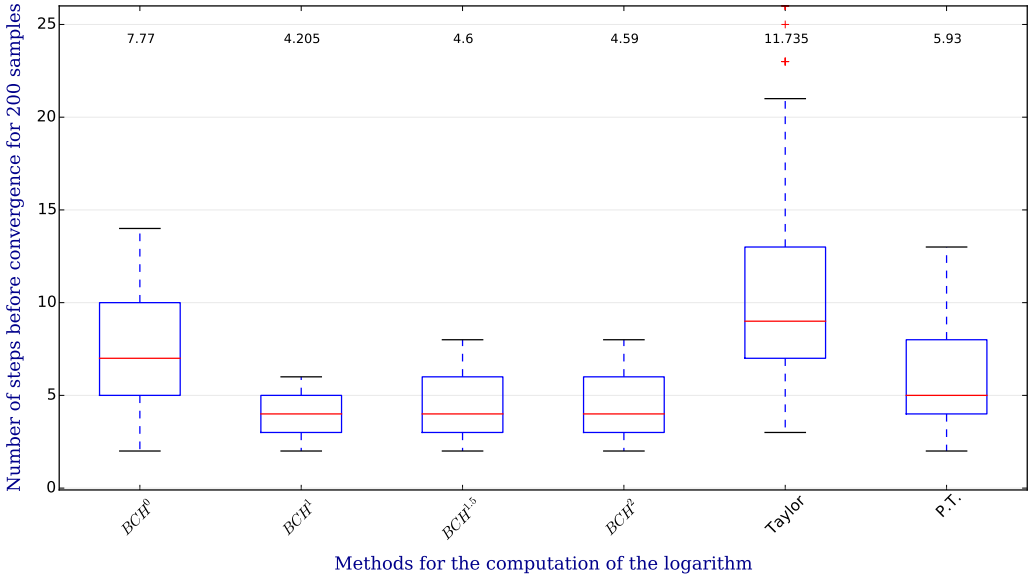



Figure 5.11: Number of steps required to obtain convergence for different methods utilized in the logarithm computation algorithm. The data set contains 200 random matrices in the Lie group $SE(2)$, with Frobenius norm between 1 and 3. On the top it is possible to visualize the mean number of step to reach the convergence for each method.

## 5.4 Lie Logarithm computation for $SE(2)$

This section is devoted the results of the numerical methods developed in chapter 4. Different numerical methods for the computation of the Lie logarithm are compared when a ground truth is available, i.e. in the finite dimensional case. We consider a data set of 200 random matrices in the Lie group $SE(2)$ with their respective logarithm in $\mathfrak{se}(2)$ computed using the closed form presented in chapter 3.

For each of the method considered and for each of the random matrix we always have convergence up to a precision of $10^{-12}$ before the $20^{\text{th}}$ iteration. In particular, in figure 5.11 we can see the average number of iterations required to reach the solution with a precision of

$10^{-4}$ for each of the numerical method considered. Each box represents 200 random matrices in $SE(2)$ with Frobenius norm uniformly selected between 1 and 3.

As obtained for the numerical computations of the Lie log-composition, results obtained with parallel transport are comparable with the results obtained with the truncated BCH of degree 1. Also in this case, an increase in the degree of the truncated BCH does not ensure a faster convergence and the Taylor method provides the slowest algorithm. The reason for this fact is not clear are the moment and, excluding errors, it may be worthed to future investigations.

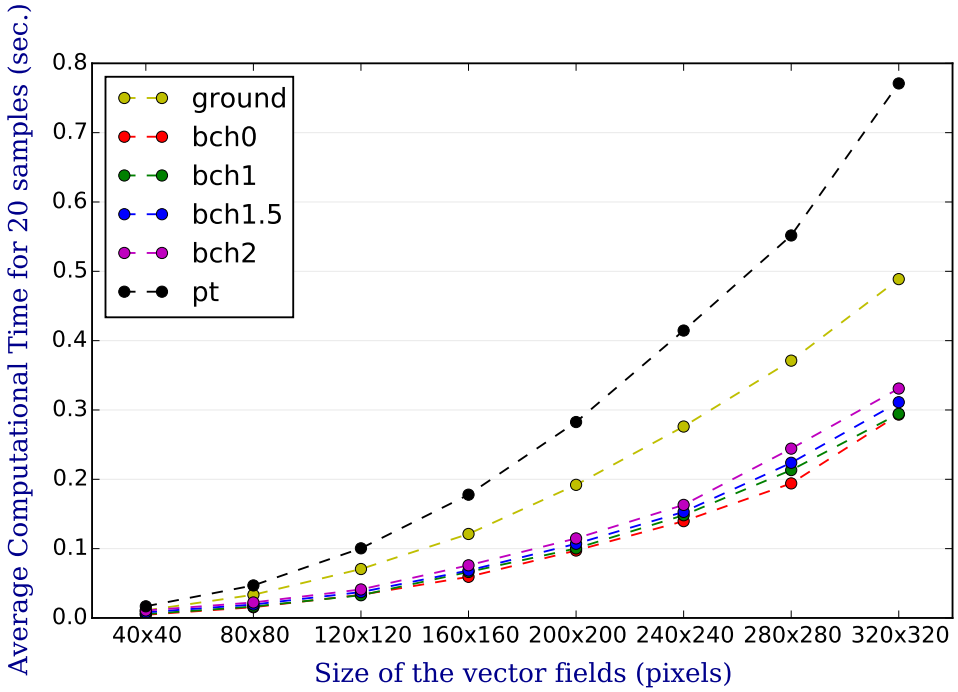



Figure 5.12: Relationships between the size of the figure (x-axes) and the computational time for a data set of 20 random generated SVF. The yellow line labeled with ground represents the time of the computation of $\exp \mathbf{u}_0 \circ \exp \mathbf{u}_1$, while the other represents the exponentiation of the numerical method for the computation of the log-composition.

## 5.5 Empirical Evaluations of the Computational Time

Last results here proposed are an empirical evaluation of the computational time. For the finite dimensional case, a dataset of 36000 random generated matrix with norm between 0.0 and 2.0 has been utilized to measure the empirical computational time of each of the numerical methods for the computation of the log-composition here presented. Computations are performed with a Mac book pro 2014, 16Gb Ram. Sum of the computational time for the whole data set are give in seconds:

| Ground | $\text{BCH}^0$ | $\text{BCH}^1$ | $\text{BCH}^{1.5}$ | $\text{BCH}^2$ | Taylor | p.t. |
|---|---|---|---|---|---|---|
| 1.07402015 | 0.18845153 | 0.57751322 | 1.24413943 | 1.78752184 | 0.77354765 | 2.26586294 |

The first column provides the computational time of the ground truth, i.e. the time to

compute the closed form for the computation of $dr_0 \oplus dr_1$. As expected the burden time to compute the $\mathrm{BCH}^0$ for a dataset of size 36000, is the lowest since it consists simply in a sum, while the computation of the parallel transport is four time slower than the $\mathrm{BCH}^1$, since it involves three times the computation of the Lie exponential and two times the resampling.

When dealing with SVF, the computational time strongly depends on the size of the vector field involved. In figure 5.12 we show the relations between the size of the vector field and the increase in the computational time for a data set of 20 random generated SVF.

# Chapter 6

# Conclusions

In this research we have formally defined the mathematical concept of Lie log-composition and we have presented the limitations of the numerical methods for its computation obtained with truncations of the BCH formula. These limitations are the starting point of the research for BCH-free numerical methods. One of these, based on the geometrical concept of parallel transport is presented here for the first time, and formally proved for element of the general Lie algebra.

This method is compared with the truncated BCH, both for elements belonging to the the Lie algebra of rigid body transformations (where also another BCH-free numerical method based on the Taylor expansion is available) and for stationary velocity fields (SVF).

The possible applications of efficient numerical methods for the computation of the Lie log-composition in medical imaging are listed in section 1.3. One of these, the computation of the Lie logarithm based on the algorithm presented in [BO08b], is investigated in chapter 4.

Results show that the Lie log-composition computed with the parallel transport method improves the $\text{BCH}^0$, and gets, in general close to the $\text{BCH}^1$. At an higher computational cost it enable to avoid the computation of the Jacobian involved in the $\text{BCH}^1$, obtaining results more precise than the $\text{BCH}^0$ when the vectors involved are considered have norm in a limited range.

Results show also that for medical applications is not recommendable to utilize numerical approximations based on $\text{BCH}^k$ for $k \geq 2$. The method that provides best results both on synthetic and real data on a wider range of SVF, is the $\text{BCH}^1$, that involves one nested Lie bracket. The parallel transport is the first choice (and the unique at the moment) when it is preferable to avoid the computation of the Lie bracket.

## 6.1 Further Researches

Since this thesis has both numerical and theoretical aspects, improvements could be reached with a better understanding of both sides.

### 6.1.1 Numerical Computations

Numerical computations on real data for small SVF, provide better results for parallel transport than for any other available numerical method (see figure 5.10) when the norm of the elements involved are smaller than 0.0227. The reason why this does not happen for synthetic data may be due to the fact that the Jacobian determinant of SVF are not always positive, when in particular for SVF with small norm. This may lead to consider better way to generating random SVF than the one proposed in this research.

To compute the Lie exponential and the Lie logarithm we always used the scaling and squaring and the inverse scaling and squaring algorithms, as originally proposed by Arsigny in 2006 [ACPA06]. There are other options available that, when applied to the log-composition, could improve the computational time. These are based for example on the Euler method, the midpoint method, the modified Euler method and Runge-Kutta of order 4. Investigations in this direction are currently in progress.

The proof of the convergence for the BCH in Dynkin's paper is based on the power series expansion of the exponential and of the logarithm (equation (2.2) and (2.3)) but these expansions, unless using the function $\mathcal{V}$ proposed in section 3.2.2 are not well defined, and have never been proved for the Lie algebra of diffeomorphisms. Numerical tests, on which we are currently working, seem to show that the Lie exponential computed with the expansion in power series converges for SVF.

### 6.1.2 Theoretical Formulas

Certainly not all of the possibilities provided by the application of the concept of parallel transport for the computation of the Lie log-composition have been exploited. The formula (2.14), presented at the end of chapter 2, can still be improved, for example finding strategies to reduce the number of underpinning assumptions and introducing some numerical techniques to compute the Affine exponential. When this tool will be available, formula (2.14) could be reformulated as

$$\mathbf{u} \oplus \mathbf{v} \simeq \exp_{\exp(\mathbf{u})}(\mathbf{v}) - e$$

or, in a symmetric version, as

$$\mathbf{u} \oplus \mathbf{v} \simeq \frac{1}{2}\big(\exp_{\exp(\mathbf{u})}(\mathbf{v}) + \exp_{\exp(\mathbf{v})}(\mathbf{u})\big) - e$$

Another BCH-free formula, not based on the parallel transport, could be obtained finding a way to extend the Taylor expansion proposed for $SE(2)$ in section 3.1 to SVF.

A third one, on which preliminary tests showed promising results, is obtained using the accelerating convergence series [CVZ00] on the series expansion of the Lie exponential and the Lie logarithm. This is based on the assumption that Lie logarithm and Lie exponential are analytic also for SVF.

# Bibliography


[ACPA06] Vincent Arsigny, Olivier Commowick, Xavier Pennec, and Nicholas Ayache. A log-euclidean framework for statistics on diffeomorphisms. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2006*, pages 924–931. Springer, 2006.

[AFPA06] Vincent Arsigny, Pierre Fillard, Xavier Pennec, and Nicholas Ayache. Log-Euclidean metrics for fast and simple calculus on diffusion tensors. *Magnetic Resonance in Medicine*, 56(2):411–421, August 2006.

[APA06] Vincent Arsigny, Xavier Pennec, and Nicholas Ayache. Bi-invariant means in lie groups. application to left-invariant polyaffine transformations. In *Research Report RR-5885, 2006. inria-00071383.* 2006.

[Arn66] Vladimir Arnold. Sur la géométrie différentielle des groupes de lie de dimension infinie et ses applications à l'hydrodynamique des fluides parfaits. In *Annales de l'institut Fourier*, volume 16, pages 319–361. Institut Fourier, 1966.

[Arn98] Vladimir Arnold. *Topological methods in hydrodynamics*, volume 125. Springer Science & Business Media, 1998.

[Arn06] Vladimir Arnold. Ordinary differential equations. translated from the russian by roger cooke. second printing of the 1992 edition. universitext, 2006.

[Art11] Michael Artin. *Algebra. 2nd.* Boston: Prentice-Hall, 2011.

[BBHM13] Martin Bauer, Martins Bruveris, Philipp Harms, and Peter W Michor. Geodesic distance for right invariant sobolev metrics of fractional order on the diffeomorphism group. *Annals of Global Analysis and Geometry*, 44(1):5–21, 2013.

[BHM10] Martin Bauer, Philipp Harms, and Peter W Michor. Sobolev metrics on shape space of surfaces in n-space. *Arxiv preprint arXiv:1009.3616*, 2010.

[BMTY05] M Faisal Beg, Michael I Miller, Alain Trouvé, and Laurent Younes. Computing large deformation metric mappings via geodesic flows of diffeomorphisms. *International journal of computer vision*, 61(2):139–157, 2005.

[BO08a] Matias Bossa and Salvador Olmos. A New Algorithm for the Computation of the Group Logarithm of Diffeomorphisms. In Xavier Pennec and Sarang Joshi, editors, *Second International Workshop on Mathematical Foundations of Computational Anatomy - Geometrical and Statistical Methods for Modelling Biological Shape Variability*, New York, USA, 2008.

[BO08b] Matias Bossa and Salvador Olmos. A new algorithm for the computation of the group logarithm of diffeomorphisms. In *2nd MICCAI Workshop on Mathematical Foundations of Computational Anatomy*, 2008.

[BZO08] Matias Bossa, Ernesto Zacur, and Salvador Olmos. Algorithms for computing the group exponential of diffeomorphisms: Performance evaluation. In *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW'08. IEEE Computer Society Conference on*, pages 1–8. IEEE, 2008.

[Car76] M. Do Carmo. *Differential geometry of curves and surfaces*, volume 2. Prentice-hall Englewood Cliffs, 1976.

[Car92] M. Do Carmo. *Riemannian geometry*. 1992.

[CVZ00] Henri Cohen, Fernando Rodriguez Villegas, and Don Zagier. Convergence acceleration of alternating series. *Experimental mathematics*, 9(1):3–12, 2000.

[Dyn00] EB Dynkin. Calculation of the coefficients in the campbell–hausdorff formula. *Selected Papers of EB Dynkin with Commentary. Originally in Institute of Mathematics Moscow State University, Acad a.n. Kormologorow, submitted 1947*, 14:31, 2000.

[EM70] David G Ebin and Jerrold Marsden. Groups of diffeomorphisms and the motion of an incompressible fluid. *Annals of Mathematics*, pages 102–163, 1970.

[FF97] Nick C Fox and Peter A Freeborough. Brain atrophy progression measured from registered serial mri: validation and application to alzheimer's disease. *Journal of Magnetic Resonance Imaging*, 7(6):1069–1075, 1997.

[Gal11] Jean Gallier. *Geometric methods and applications: for computer science and engineering*, volume 38. Springer Science & Business Media, 2011.

[Gra88] Janusz Grabowski. Free subgroups of diffeomorphism groups. *Fundamenta Mathematicae*, 2(131):103–121, 1988.

[GWRNJ12] Serge Gauthier, Liyong Wu, Pedro Rosa-Neto, and Jianping Jia. Prevention strategies for alzheimer's disease. *Translational neurodegeneration*, 1(1):1–4, 2012.

[Hal15] Brian Hall. *Lie groups, Lie algebras, and representations: an elementary introduction*, volume 222. Springer, 2015.

[HSSE09] Darryl D Holm, Tanya Schmah, Cristina Stoica, and David CP Ellis. *Geometric mechanics and symmetry: from finite to infinite dimensions*. Oxford University Press London, 2009.

[Hun07] J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing In Science & Engineering*, 9(3):90–95, 2007.

[HWS+] Zhiwu Huang, Ruiping Wang, Shiguang Shan, Xianqiu Li, and Xilin Chen. Log-euclidean metric learning on symmetric positive definite manifold with application to image set classification.

[ISNC03] Luis Ibanez, William Schroeder, Lydia Ng, and Josh Cates. The itk software guide. 2003.

[JBF+08] Clifford R Jack, Matt A Bernstein, Nick C Fox, Paul Thompson, Gene Alexander, Danielle Harvey, Bret Borowski, Paula J Britson, Jennifer L Whitwell, Chadwick Ward, et al. The alzheimer's disease neuroimaging initiative (adni): Mri methods. *Journal of Magnetic Resonance Imaging*, 27(4):685–691, 2008.

[JOP+ ] Eric Jones, Travis Oliphant, Pearu Peterson, et al. SciPy: Open source scientific tools for Python, 2001–. [Online; accessed 2015-08-03].

[Kir08] Alexander Kirillov. *An introduction to Lie groups and Lie algebras*, volume 113. Cambridge University Press Cambridge, 2008.

[KMN00] Arkady Kheyfets, Warner A Miller, and Gregory A Newton. Schild's ladder parallel transport procedure for an arbitrary connection. *International Journal of Theoretical Physics*, 39(12):2891–2898, 2000.

[Kne51] MS Knebelman. Spaces of relative parallelism. *Annals of Mathematics*, pages 387–399, 1951.

[KO89] S Klarsfeld and JA Oteo. The baker-campbell-hausdorff formula and the convergence of the magnus expansion. *Journal of physics A: mathematical and general*, 22(21):4565, 1989.

[KW08] Boris Khesin and Robert Wendt. *The geometry of infinite-dimensional groups*, volume 51. Springer Science & Business Media, 2008.

[LAP11] Marco Lorenzi, Nicholas Ayache, and Xavier Pennec. Schild's ladder for the parallel transport of deformations in time series of images. In *Information Processing in Medical Imaging*, pages 463–474. Springer, 2011.

[Lee12] John Lee. *Introduction to smooth manifolds*, volume 218. Springer Science & Business Media, 2012.

[Les83] JA Leslie. A lie group structure for the group of analytic diffeomorphisms. *Boll. Un. Mat. Ital. A (6)*, 2:29–37, 1983.

[LP13] Marco Lorenzi and Xavier Pennec. Geodesics, parallel transport & one-parameter subgroups for diffeomorphic image registration. *International journal of computer vision*, 105(2):111–127, 2013.

[LP14a] Marco Lorenzi and Xavier Pennec. Discrete Ladders for Parallel Transport in Transformation Groups with an Affine Connection Structure. In Frank Nielsen, editor, *Geometric Theory of Information*, Signals and Communication Technology, pages 243–271. Springer, 2014.

[LP14b] Marco Lorenzi and Xavier Pennec. Efficient parallel transport of deformations in time series of images: from schild's to pole ladder. *Journal of Mathematical Imaging and Vision*, 50(1-2):5–17, 2014.

[MA70] J Marsden and R Abraham. Hamiltonian mechanics on lie groups and hydrodynamics. *Global Analysis,(eds. SS Chern and S. Smale), Proc. Sympos. Pure Math*, 16:237–244, 1970.

[MHM+11] JR McClelland, S Hughes, M Modat, A Qureshi, S Ahmad, DB Landau, S Ourselin, and DJ Hawkes. Inter-fraction variations in respiratory motion models. *Physics in medicine and biology*, 56(1):251, 2011.

[MHSK] J. R. McClelland, D. J. Hawkes, T. Schaeffter, and A. P. King. Respiratory motion models: A review. *Medical Image Analysis*, 17(1):19–42, 2015/04/01.

[Mic80] P Michor. Manifolds of smooth maps, ii: the lie group of diffeomorphisms of a non-compact smooth manifold. *Cahiers de Topologie et Géométrie Différentielle Catégoriques*, 21(1):63–86, 1980.

[Mil82] John Milnor. On infinite dimensional lie groups. *Preprint, Institute for Advanced Study, Princeton*, 1982.

[Mil84a] J Milnor. Remarks on infinite-dimensional lie groups, in 'relativity, groups and topology, ii'(les houches, 1983), 1007–1057, 1984.

[Mil84b] John Milnor. Remarks on infinite-dimensional lie groups. In *Relativity, groups and topology. 2*. 1984.

[MRT+10] Marc Modat, Gerard R Ridgway, Zeike A Taylor, Manja Lehmann, Josephine Barnes, David J Hawkes, Nick C Fox, and Sébastien Ourselin. Fast free-form deformation using graphics processing units. *Computer methods and programs in biomedicine*, 98(3):278–284, 2010.

[MT13] Carlo Mariconda and Alberto Tonolo. *Calcolo discreto: Metodi per contare*. Apogeo Editore, 2013.

[MTW73] Charles W Misner, Kip S Thorne, and John Archibald Wheeler. *Gravitation*. Macmillan, 1973.

[Nee06] KH Neeb. Infinite dimensional lie groups, 2005 monastir summer school lectures. *Lecture Notes January*, 2006.

[OKC92] V Yu Ovsienko, BA Khesin, and Yu V Chekanov. Integrals of the euler equations of multidimensional hydrodynamics and superconductivity. *Journal of Soviet Mathematics*, 59(5):1096–1101, 1992.

[Omo70] Hideki Omori. On the group of diffeomorphisms on a compact manifold. In *Proc. Symp. Pure Appl. Math., XV, Amer. Math. Soc*, pages 167–183, 1970.

[PCL+15] Ferran Prados, Manuel Jorge Cardoso, Kelvin K Leung, David M Cash, Marc Modat, Nick C Fox, Claudia AM Wheeler-Kingshott, Sebastien Ourselin, Alzheimer's Disease Neuroimaging Initiative, et al. Measuring brain atrophy with a generalized formulation of the boundary shift integral. *Neurobiology of aging*, 36:S81–S90, 2015.

[PL+11] Xavier Pennec, Marco Lorenzi, et al. Which parallel transport for the statistical analysis of longitudinal deformations. In *Proc. Colloque GRETSI*. Citeseer, 2011.

[Sch10] Rudolf Schmid. Infinite-dimensional lie groups and algebras in mathematical physics. *Advances in Mathematical Physics*, 2010, 2010.

[SDP13] A. Sotiras, C. Davatzikos, and N. Paragios. Deformable medical image registration: A survey. *Medical Imaging, IEEE Transactions on*, 32(7):1153–1190, July 2013.

[Ser09] Jean-Pierre Serre. *Lie algebras and Lie groups: 1964 lectures given at Harvard University*. Springer, 2009.

[Sze94] Richard Szeliski. Image mosaicing for tele-reality applications. In *Applications of Computer Vision, 1994., Proceedings of the Second IEEE Workshop on*, pages 44–53. IEEE, 1994.

[Ver14] Tom Vercauteren. Ideas and notes on globally consistent registration frompairwise registration results. technical memo. Pre-print, 2014.

[VPM+06] Tom Vercauteren, Aymeric Perchant, Grégoire Malandain, Xavier Pennec, and Nicholas Ayache. Robust mosaicing with correction of motion distortions and tissue deformations for in vivo fibered microscopy. *Medical image analysis*, 10(5):673–692, 2006.

[VPPA07] Tom Vercauteren, Xavier Pennec, Aymeric Perchant, and Nicholas Ayache. Non-parametric diffeomorphic image registration with the demons algorithm. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2007*, pages 319–326. Springer, 2007.

[VPPA08] Tom Vercauteren, Xavier Pennec, Aymeric Perchant, and Nicholas Ayache. Symmetric log-domain diffeomorphic registration: A demons-based approach. In Dimitris N. Metaxas, Leon Axel, Gabor Fichtinger, and Gábor Székely, editors, *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2008, 11th International Conference, New York, NY, USA, September 6-10, 2008, Proceedings, Part I*, volume 5241 of *Lecture Notes in Computer Science*, pages 754–761. Springer, 2008.

[War13] Frank W Warner. *Foundations of differentiable manifolds and Lie groups*, volume 94. Springer Science & Business Media, 2013.

[Wig60] Eugene P Wigner. The unreasonable effectiveness of mathematics in the natural sciences. richard courant lecture in mathematical sciences delivered at new york university, may 11, 1959. *Communications on pure and applied mathematics*, 13(1):1–14, 1960.

[Woj94] Wojciech Wojtyński. One-parameter subgroups and the bch formula. *Studia Mathematica*, 111(2):163–185, 1994.

[YC06] Lexing Ying and Emmanuel J Candès. The phase flow method. *Journal of Computational Physics*, 220(1):184–215, 2006.

[You10] Laurent Younes. *Shapes and diffeomorphisms, volume 171 of Applied Mathematical Sciences*. Springer, 2010.