Mechanizing Lagrangian Mechanics in Isabelle/HOL

Dawson Silkenat

4th Year Project Report
Artificial Intelligence and Mathematics
School of Informatics
University of Edinburgh

2022

Abstract

This project attempts to formalise Lagrangian mechanics with a focus on the ability to derive equations of motion and conservation laws. We do this using the proof assistant Isabelle/HOL, which verifies the correctness of our proofs. Our formalisation of Lagrangian mechanics was largely inspired by Sussman and Wisdom's Structure and Interpretation of Classical Mechanics [22], although there are significant differences between their approach to Lagrangian mechanics and the approach taken in this paper.

Acknowledgements

I would like to thank my supervisor, Jacques Fleuriot, for his guidance and for suggesting this project. I would also like to thank Richard Schmoetten and Jake Palmer for their advice and support.

Table of Contents

Chapter 1

Introduction

The goals of this project were to create a locale formalising Lagrangian mechanics, use this locale to derive equations of motion, and prove Noether's theorem. In this paper we discuss how we approached these problems. We believe we have successfully achieved these goals.

This paper is organised in the following manner:

Chapter 2 discusses background knowledge to help contextualise the rest of this paper. We also discuss related work.

Chapter 3 discusses our formalisation of Lagrangian mechanics as an Isabelle locale.

Chapter 4 discusses the general approach taken to deriving equations of motion from our formalisation of Lagrangian mechanics. We then discuss five examples of systems for which we have derived equations of motion.

Chapter 5 discusses our proofs of two conservation laws in Lagrangian mechanics, and provide an example of applying each.

Chapter 6 contains concluding remarks.

The main contributions of our project were:

- An Isabelle locale formalising Lagrangian mechanics, designed so that it can be used to derive equations of motion for specific systems and derive conservation laws.
- The derivation of equations of motion for five example systems of varying degrees of complexity using the aforementioned Isabelle locale.
- the proofs and applications of conservation laws, most notably Noether's theorem.

There exists some overlap between the first two points on this list and the work of Guan et al [7], however the approaches taken to arrive at these results differ drastically. Our most important contribution was the proofs and applications of conservation laws which are, to the best of our knowledge, the only such formalisation verified by a proof assistant.

Chapter 2

Background

2.1 Mechanics

Mechanics is a branch of physics describing the motion of and interaction between particles in a system. The first, and by far the most famous, formulation of mechanics is that of Isaac Newton in the Principia [15]. This formulation is known as Newtonian Mechanics, which remains widely used and provides very accurate results for a large number of problems.

The importance of mechanics is difficult to overstate. It has applications in many fields, perhaps most notably in several branches of engineering. Mechanics has also been the driving force behind some major advancements in mathematics and is a core part of any physicist's education.

Lagrangian mechanics offers an alternative way of describing the same laws of nature as Newtonian mechanics. In this formulation we start with a configuration space, which consists of all possible arrangements of the particles in a system. We can then specify the motion of the system as a function from time to the configuration space. Each configuration can be labelled with a coordinate according to some coordinate system of our choosing. We call this coordinate system the generalized coordinates. A coordinate path function can then be defined as a map from time to the coordinates representing the configuration of our system at that point in time.

Not all possible coordinate paths describe motion which can occur. To distinguish those that do we define the action of a coordinate path q from time t_1 to t_2 as

$$S[q](t_1,t_2) = \int_{t_1}^{t_2} L \circ \Gamma[q]$$

where $\Gamma[q]$ is a function from time to a tuple consisting of the time, the value of q at that time, and the values of a number of derivatives of q at that time, while L is a real valued function known as the Lagrangian of the system. The paths which accurately

describe the motion of the system are those for which $S[q]$ is stationary[1] with respect to variation in q.

To use this method to determine q, we must know a Lagrangian L which describes the system. An important result of Lagrangian mechanics is that for any system of particles where force can be defined as the negative derivative of potential energy and is independent of the derivative of q, which we will denote $\dot{q}$, a Lagrangian can be defined as the difference between the kinetic energy and potential energy of the system. This is known as Hamilton's principle.

If we know a Lagrangian can be defined as the difference between the kinetic energy and potential energy of the system then we do not need to rely upon the formula for action to determine which coordinate path describes the motion of the system. For a given path q, such a Lagrangian will depend only on time, the value of q at that time, and the first derivative of q at that time. The coordinate path which describes the motion of the system must satisfy

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = 0$$

for each component of q_i of q. This produces a system of equations known as the Euler-Lagrange equations.

While it may not always possible to define a Lagrangian as the difference between kinetic and potential energy, this approach is so ubiquitous that some texts take the difference between kinetic and potential energy to be the definition of the Lagrangian [5, 14].

The Euler-Lagrange equations do not depend on our choice of coordinate system. This gives us significant freedom when deciding our coordinate system; the method we use to determine valid paths will not be affected. This is a significant advantage over Newtonian mechanics, where force being a vector quantity often restricts which coordinate system can realistically be used.

Another advantage of Lagrangian mechanics over Newtonian mechanics is the ease with which constraints on the system can be expressed. In Newtonian mechanics this typically involves considering the constraints as the result of particular forces. This can make the system significantly more complex to work with. With Lagrangian mechanics we can select a coordinate system in which violating the constraint isn't possible.

A useful property of Lagrangian mechanics is its ability derive the conservation laws of a given system using Noether's theorem. Noether's theorem states that for every continuous symmetry of the Lagrangian there exists an associated conservation law. This is incredibly important since conservation laws play a fundamental part in our understanding of how the world works.

[1] The definition of stationary in this context is similar to that of a stationary point, except the variation is a small change in a function rather than a variable.

2.2 Isabelle/HOL

In theory a formal proof of a statement will show every step of the process, every assumption made and every axiom used. It leaves no room for doubt about the truth of the statement provided each step is properly justified. In practice, formal proofs can take significant effort both from the writer and reader to verify correctness and usefulness of a proof, which many find uninteresting.

For this reason most proofs are not formal proofs, but rather an argument which is merely considered sufficiently formal while remaining (relatively) concise and understandable. Such an argument may contain errors or unstated assumptions which are more difficult to detect than in a formal proof. Such mistakes and hidden assumptions have been found in the work of many great mathematicians.

Isabelle/HOL is a generic proof assistant which aids the writing of formal proofs and verification of their correctness. Such a proof assistant makes the idea of formal proof more tractable.

Isabelle proof scripts can be written in the Isar structured proof language which provides a human-readable proof script. While one of the main reasons for using Isabelle/HOL is the correctness of proofs being verified by a computer, it remains important that our proofs also be human-readable for two main reasons. The first and perhaps more important is while Isabelle/HOL can verify the correctness of a proof, it cannot understand what we are trying to prove. A mistake in the formulation of the claim may result in a correct proof of the wrong statement. The second reason is a human-readable script is easier to maintain and build upon. Understanding why a particular theorem is true can be as important as knowing it is true, similar techniques may prove useful in proving other theorems.

Isabelle has a number of automated tools which both make the process of writing the proof easier and the resulting proofs easier to read. Automated proof methods such as simp, auto and blast allow us to specify the goal we are trying to prove and the theorems we believe are required to prove it using the specific routines the methods implement, while letting the computer figure out the formal proof. This allows us to write our proofs in a style that more closely matches a typical pen-and-paper proof while Isabelle keeps track of the fully formal proof.

Sledgehammer is another very powerful automated tool. It tries a number of automated theorem provers and relevant theorems to search for a proof of the current goal. This can be useful when we believe the current to be provable, but do not know precisely which theorems are needed to prove it. If it doesn't find a proof, it may be possible to guide to one by suggesting some useful theorems or simplifying the goal it is trying to prove by breaking it into smaller steps.

Isabelle/HOL includes a large library of relevant mathematics which has already been proven and verified, making the task of formalising Lagrangian mechanics significantly simpler. There also exists the archive of formal proofs, which contains a vast number of Isabelle proof scripts freely available to download and use. This can be a useful resource for finding theorems and definitions related to our task.

2.3 Related Work

David Hilbert set forth 23 problems which he believed the study of would lead to advancements in mathematics and science [8]. Since their publication these problems have been influential in the progression of mathematics. Sixth among them was a "Mathematical Treatment of the Axioms of Physics".

Following Hilbert's sixth problem there have been attempts to formalise some fields of physics [6]. This includes interest in and attempts at a formal treatment of classical mechanics [13, 18]. Of particular contention in the formalisation of Newtonian mechanics is the appropriate treatment of force, with some even arguing that any formal definition will only allow for circular reasoning [5, volume 1, chapter 12]. We did not find attempts to formalise Lagrangian mechanics outside of proof assistants, perhaps because Lagrange's original formulation was already very mathematical.

Use of a proof assistant for formalising physics seems a natural choice, however this does not appear to have been very well explored. Most interest in this area appears to be formalisations of special relativity [11, 16, 19]. While each of these formalisations is intriguing, there are many fields of physics besides special relativity which also deserve to be formalised. Another area of interest is the formalisation of control systems for various robots [4, 10, 12, 17]. These formalisations make use of equations derived using classical mechanics, but do not themselves derive these equations from principles of physics. While useful for their intended purpose, we do not consider these sufficiently general to be formalisations of physics.

Our work is most closely related to that of Guan et al [7]. They formalised significant theorems in variational calculus, then used those theorems to show conditions under which the Euler-Lagrange equations are the solutions to the variational problem in the principle of least action. They conclude by demonstrate how to apply their formalisation to an example problem. This work is significant, but far from comprehensive. For instance, they have not begun to formalise conservation laws.

Chapter 3

Lagrangian Mechanics

In this chapter we discuss the decisions and challenges encountered in designing our formalisation of Lagrangian mechanics in Isabelle. We start with a discussion regarding the axioms of our formalisation in section 3.1. We then discuss the representation of derivatives in section 3.2 and the Lagrangian in section 3.3. We conclude this chapter by discussing how we put these pieces together for our formalisation section 3.4.

Throughout this chapter we will use Isabelle's `euclidean_space` type to represent vectors. Euclidean spaces have been used since the ancient Greeks to understand the world. The axioms Isabelle's `euclidean_space` type requires our vectors satisfy encapsulate this geometry, which is correct for classical mechanics.

Throughout this paper we will use the following notation: q will represent the coordinate path describing the motion of the system. $\dot{q}$ will represent the time derivative of this coordinate path. In equations we will use L to refer to the Lagrangian.

3.1 Selecting Axioms

We choose the Euler-Lagrange equations to be the foundations of our formalisation of Lagrangian mechanics. This may seem a strange choice as Lagrangian mechanics is based on the principle of least action, from which the Euler-Lagrange equations are derived. We choose not to make the principle of least action our starting point because the derivation of Euler-Lagrange equations relies on mathematics for which we have some intuition but no formal knowledge. Gaining sufficient understanding to prove the Euler-Lagrange equations would be a significant and time consuming undertaking.

By choosing the Euler-Lagrange equations over the principle of least action it becomes feasible to prove some of the more interesting results in Lagrangian mechanics within the time available for the project. We focus our efforts towards proving conservation laws, discussed in chapter 5. Of particular interest is Noether's theorem, whose proof and application turned out to be particularly interesting and challenging. Had we started from the principle of least action we very much doubt time would have permitted any attempt to prove Noether's theorem.

This choice does lead to some loss of generality. The Euler-Lagrange equations only work if the Lagrangian depends only on t, q, and $\dot{q}$, while principle of least action is a statement that can be made for a Lagrangian which depends on any number of derivatives of q.

We do not consider this loss of generality to be very significant. When using Lagrangian mechanics it is very common for the Lagrangian to be the difference between the kinetic energy, which depends on q and $\dot{q}$, and the potential energy, which depends only on q. This equation for the Lagrangian is so common that some treat it as a definition for the Lagrangian. When the Lagrangian is defined in this manner it does not depend on derivatives of q beyond the first, so the Euler-Lagrange equations can be used. On the other hand, the principle of least action is practically never used outside the derivation of the Euler-Lagrange equations.

3.2 Derivatives

Having decided to treat the Euler-Lagrange equations as the foundation of our formalisation of Lagrangian mechanics in section 3.1, we must decide how we represent the relevant derivatives. There are two forms of derivative we must consider: the derivative of a function of a single real variable and the partial derivative.

It would be possible to write a single definition which encapsulates both derivatives since the definition of the partial derivative for a function of a single variable is identical to that of the ordinary derivative. We chose to use distinct definitions as we believe it makes our intent when taking a derivative more clear. A similar choice is made in pen and paper mathematics

3.2.1 Ordinary Derivative

We start by considering the derivative of a function of a single real variable, which we will use to represent the time derivative that appears in the Euler-Lagrange equations and that $\dot{q}$ is the derivative of q.

In Isabelle's analysis library there are multiple definitions of the derivative, all of which depend on the definition of `has_derivative`. At first using `has_derivative` directly seems a good choice to represent our normal notion of the derivative, however this is not the case. In Isabelle `has_derivative` represents the Fréchet derivative and has the following definition.

```
definition has_derivative :: "('a::real_normed_vector ⇒
    'b::real_normed_vector) ⇒ ('a ⇒ 'b) ⇒ 'a filter ⇒ bool"
  where "(f has_derivative f') F <->
    bounded_linear f' ∧ ((λy. ((f y - f (Lim F (λx. x))) - f' (y -
   Lim F (λx. x))) /R norm (y - Lim F (λx. x))) -> 0) F"
```

This definition can be interpreted as saying f has derivative f' if

$$\lim_{y \to x} \frac{f(y) - f(x) - f'(y-x)}{|y-x|} = 0. \tag{3.1}$$

We can see this is not the desired form of the derivative using a technique known as dimensional analysis. This is a technique where the quantity (such as distance or time) for each term in an equation is considered rather than the numerical values. We refer to this quantity as the dimension of the term. Operations such as addition, subtraction, and equality only make sense between terms of the same dimension. Multiplication between terms of different dimensions are allowed, and the dimension of the result is the product of the dimensions. Multiplication can include division, where the units of a reciprocal are the reciprocal of the units

To illustrate this consider distance, speed and time. It would not make sense to say a speed is equal to a distance or add speed to a distance because speed and distance have different units. On the other hand, it is perfectly sensible to multiply a speed by time to get the distance travelled, because the product of the units of speed and time are the same as the units of distance.

Now consider the standard definition of the derivative.

$$f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h} \tag{3.2}$$

If we are taking the derivative with respect to time, h must be measured in units of time. $f(x+h) - f(x)$ will have the same units as $f(x)$ since subtraction does not change the units, while dividing by h will result in a change in dimension. We then see $f(x)$ and $f'(x)$ have different dimensions. In the definition of the Fréchet derivative we see that f' is subtracted from f.

If we would like f' to be the derivative of f with respect to time, then the dimensions of the two cannot agree. We would therefore be preforming a nonsensical operation, which leads us to the conclusion that there must be some mistake in how we have applied the Fréchet derivative to our situation.

Another choice of derivative is `has_vector_derivative`, defined in Isabelle as

```
definition has_vector_derivative :: "(real ⇒
  'b::real_normed_vector) ⇒ 'b ⇒ real filter ⇒ bool"
  where (f has_vector_derivative f') net
    ↔ (f has_derivative (λx. x *ᴿ f')) net
```

Using equation 3.1 we can see this Isabelle definition represents the statement f has derivative f' if

$$\lim_{y \to x} \frac{f(y) - f(x) - (y-x)f'(x)}{|y-x|} = 0. \tag{3.3}$$

The term $(y-x)$ will have a time dimension which causes $(y-x)f'(x)$ to have the same dimension as f, so this equation valid from the perspective of dimensional analysis.[1]

[1] This does not guarantee correctness, it is very possible to write nonsense equations which are dimensionally correct.

Additionally, with some manipulation it can be shown this form of the derivative is equivalent to more traditional definitions of the derivative seen in equation 3.2.

While this is certainly the form of the derivative we desire we will largely be using `vector_derivative` rather than `has_vector_derivative`.

This is because `has_vector_derivative` is a predicate which must be provided with both the function f and its derivative f'. To use this predicate to express second derivatives we would need to fix two functions, f and f', then use `has_vector_derivative` to assume that the value of f' is the derivative of f for every value of t. It would then be possible to fix a third function f'' and assume its value is the derivative of f' at every value of t. This process is convoluted, and, while it may be possible to write readable definitions using this approach, there is an easier method.

We use `vector_derivative` because it is a function which takes a function f as an argument and whose value is the derivative f', provided such a value exists. This makes it very easy to define second derivatives. For a given function f we can define a new function f' whose value at each time t is the derivative of f at t. This f' will itself be a function, so we can apply `vector_derivative` to it to get the second derivative of f.

The disadvantage of using `vector_derivative` is that it can be applied to functions for which a derivative does not exist. `vector_derivative` is still defined in this case, it is required to be since functions and predicates in Isabelle are assumed to be total, but its value is meaningless and cannot be used to show desired properties. This will require us to include differentiability assumptions to make sure the derivatives we are defining actually exist.

3.2.2 Partial Derivative

We now consider the partial derivative, which we will use to represent the partial derivatives of the Lagrangian with respect to the elements of q and $\dot{q}$. Unlike the standard derivative, there are not definitions of the partial derivative in Isabelle's analysis library. We instead find a definition through the archive of formal proofs in Abdulaziz and Paulson's formalisation of Green's theorem [1]. They define the partial derivative as the rate of change of the function with respect to the magnitude of a vector b.

$$\left.\frac{\partial F(v)}{\partial b}\right|_{v=a} = \left.\frac{\mathrm{d}F(a+(x-a\cdot b)b)}{\mathrm{d}x}\right|_{x=a\cdot b}$$

They formalise this as the following Isabelle definition.

```
definition has_partial_vector_derivative :: "('a::euclidean_space ⇒
    'b::euclidean_space) ⇒ 'a ⇒ 'b ⇒ 'a ⇒ bool"
  where "has_partial_vector_derivative F b F' a ≡ ((λx. F((a -
  ((a • b) *R b)) + x *R b )) has_vector_derivative F') (at (a • b))"
```

It should be noted that this definition is not how the partial derivative is usually defined since it takes the derivative along a vector rather than with respect to one of the variables on which the function depends. In fact this definition of the derivative is usually referred to as the directional derivative, not the partial derivative. Despite not being the version

of the derivative its name would imply, this definition is still suitable for our purposes since the partial derivative is a special case of the directional derivative.

In particular there is a one to one correspondence between the partial derivatives of a function and the directional derivatives along the standard basis vectors.[2] In the Euler-Lagrange equations we have terms of the form $\partial L/\partial q_i$ and $\partial L/\partial \dot{q}_i$, where i is a natural number indexing the element with which we are taking the derivative with respect to. Rather than use natural numbers we will use the correspondence between partial derivatives and directional derivative to index each of our partial derivatives with a basis vector.

The required basis can be acquired from the Isabelle `euclidean_space` type since its axioms require the existence of an orthonormal basis. From properties of orthonormal bases we may assume this is the standard basis.

Similar to `vector_derivative` there exists a analogous definition of `partial_vector_derivative`, which we will again favour for similar reasons.

3.3 The Lagrangian

The last piece to our formalisation of Lagrangian mechanics is a definition for the Lagrangian. Without this, there is no system specific information so our formalisation would be incapable of deriving any useful results.

There are a number of ways the Lagrangian could be defined when working with pen and paper. Since our formalisation was in Isabelle rather than on paper and we have assumed the Euler-Lagrange equations as first principles, the only sensible definition of the Lagrangian we could come up with was as a function of three variables. All other representations considered either did not allow us to take one or more of the derivatives required by the Euler-Lagrange equations or were convoluted methods of making an equivalent definition.

The three variables should represent t and the functions $q(t)$ and $\dot{q}(t)$, however the Lagrangian cannot be a function of functions since the derivatives discussed in section 3.2 only apply to functions of variables. We will therefore consider the Lagrangian to be a function of variable and evaluate the derivatives appearing in the Euler-Lagrange equations at values of our functions $q(t)$ and $\dot{q}(t)$.

3.4 Foundations Isabelle Locale

We have now established the assumptions and tools required to formalise Lagrangian mechanics in Isabelle. We split our formalisation into two locales. The first locale contains the definitions we find important for formalising Lagrangian mechanics, but none of the assumptions or lemmas. Our second locale extends the first by adding the required assumptions. By splitting our formalisation over two locales, we are able to

[2] For instance, the vectors $\begin{bmatrix}1\\0\end{bmatrix}$ and $\begin{bmatrix}0\\1\end{bmatrix}$ are the two elements of the standard basis for $\mathbb{R}^2$.

use definitions from the first locale to represent the assumptions in the second locale. This results in a much more readable formalisation.

The definitions locale must fix both the Lagrangian and q as our definitions will depend on one or both of these functions.

```
locale lagrangian_mechanics_defs =
  fixes lagrangian :: "real ⇒ 'a::euclidean_space ⇒ 'a ⇒ real"
    and q :: "real ⇒ 'a"
```

The following functions are defined in this locale, and will be important for understanding our formalisation of Lagrangian mechanics.

The function `qdot` represents $\dot{q}$ and is defined as

```
definition qdot :: "real ⇒ 'a"
  where "qdot t ≡ (vector_derivative q) (at t)"
```

The function `q_partial` represents $\partial L/\partial q_i$ and is defined as

```
definition q_partial :: "'a ⇒ real ⇒ real"
  where "q_partial base_vec t ≡ partial_vector_derivative
  (λx. lagrangian t x (qdot t)) base_vec (q t)"
```

Similarly the function `qdot_partial` represents $\partial L/\partial \dot{q}_i$ and is defined as

```
definition qdot_partial :: "'a ⇒ real ⇒ real"
  where "qdot_partial base_vec t  ≡ partial_vector_derivative
  (λx. lagrangian t (q t) x) base_vec (qdot t)"
```

Finally the function `dt_qdot_partial` represents $\mathrm{d}(\partial L/\partial \dot{q}_i)/\mathrm{d}t$ and is defined as

```
definition dt_qdot_partial :: "'a ⇒ real ⇒ real"
  where "dt_qdot_partial base_vec t ≡ vector_derivative
    (qdot_partial base_vec) (at t)"
```

Using these definitions we can now define the second locale which is our formalisation of Lagrangian mechanics.

```
locale lagrangian_mechanics = lagrangian_mechanics_defs +
  assumes "⋀t. q differentiable (at t)"
  assumes "⋀t. (λx. lagrangian t (fst x) (snd x)) differentiable at
    (q t, qdot t)"
  assumes "⋀t. (qdot_partial base_vec) differentiable (at t)"
  assumes lagrange_equation: "⋀t. ∀base_vec∈Basis. (dt_qdot_partial
    base_vec t) - (q_partial base_vec t) = 0"
```

All assumptions of this locale relate to the Euler-Lagrange equations. The first assumptions is that q is differentiable. This is required as a derivative with respect to $\dot{q}$ appears in the Euler-Lagrange equations, which would not be well defined if q where not differentiable. The second assumption is that the Lagrangian is differentiable. There are number of ways this assumptions could have been stated, not all of which are equivalent. For instance, we could have assumed every partial derivative of the Lagrangian existed. We choose to assume the Lagrangian is differentiable with respect to the tuple $(q, \dot{q})$ as we found this to be required for the multiple variable chain rule, discussed in

section 5.1. The third assumptions is that $\partial L/\partial q$ is differentiable with respect to time, which is required since this derivative appears in the Euler-Lagrange equations. The final assumption is that the Euler-Lagrange equations are satisfied, which is what makes this locale a formalisation of Lagrangian mechanics.

Chapter 4

Representing Mechanical Systems

In section 3.4 we created a locale mechanising Lagrangian mechanics where we have used the Euler-Lagrangian equations as the basic axioms. We believed the decisions discussed throughout chapter 3 resulted in this locale accurately represents the assumptions and definitions of our selected subset of Lagrangian mechanics. Prior to this project we had no experience with Lagrangian mechanics, so there existed some uncertainty that we had correctly understood the properties we are trying to formalise. One way we can establish confidence that our mechanization is accurate is to apply it to a mechanical system and derive accurate results. This would also show that our mechanisation of Lagrangian mechanics can actually be used to formally prove properties of a system rather than being an entirely abstract representation.

4.1 Structure

In section 4.2 we will derive the equations of motion for five system. Before discussing specific examples of using the Lagrangian mechanics locale to derive equations of motion it is important to understand the design decisions that have been made, and the alternatives considered.

4.1.1 Representing Systems as Extensions

We will be implementing each system as their own locale. To make use of the assumptions and definitions from the Lagrangian mechanics locale in the locale representing a particular system we must establish a relation between the two. We do this by implementing the system's locale as an extension of the Lagrangian mechanics locale, similar to how we implemented the Lagrangian mechanics locale as an extension of the Lagrangian mechanics definitions locale in section 3.4.

The main thing this extension must do is assume an equation for the Lagrangian. If we knew nothing about the equation for the Lagrangian we would not be able to use the Euler-Lagrange equations to derive information about q, making the Lagrangian mechanics locale essentially meaningless. In many cases a Lagrangian will depend on

some scalar quantities, such as mass or length. These will be fixed by the locale, and may include some assumptions about their values.

We would like to include as few assumptions regarding q as possible. If we were to make no assumptions about q then any results we derive in the locale must hold for every possible path for which the Euler-Lagrange equations are satisfied. This ideal world is not possible since, in the Lagrangian mechanics locale, we have already assumed q is once differentiable. You will see in section 4.2 that we frequently need to assume q is twice differentiable. In some cases, such as in the Kepler problem discussed in section 4.2.5, we will require additional assumptions regarding q which are not related to differentiability.

Even though we make some assumptions regarding the path function q, the results we derive remain very general by including assumptions only when they are necessary. We consider an assumption to be necessary only if the equation of motion cannot be derived without it or if a violation of that assumption would fundamentally change the system. A quite common example we encounter is that the Euler-Lagrange equations result in an expression including the second derivative of q. There is little choice but to assume q is twice differentiable, otherwise the only possibilities are that the system cannot be described using Lagrangian mechanics or that such a path does not describe the motion of the system. Neither of situations is worth considering in a locale which assumes the Euler-Lagrange equations are satisfied.

An alternative approach is to have the system locale fix a path and Lagrangian, then show that this pair satisfies the assumptions of the Lagrangian mechanics locale. This would be done using Isabelle's `interpretation`. To make this approach work, we would need the locale to specify extensive information regarding the path and Lagrangian.

One choice would be to specify specific formulas for both q and the Lagrangian. While this would allow us to show that specific path can describe the motion of the system, any results we derive will only apply to that path. We would much rather be able to show a given property holds for all paths which can describe the motion of the system.

To regain some level of generality rather than a specific equation for the path we could state some properties the path must satisfy. In the best case these assumptions are as weak as possible, in which case no generality would be lost. Proving the assumptions cannot be made weaker would likely be quite challenging, significantly more so than deriving the properties by writing the locale as an extension. Not including such a proof would leave doubt regarding the generality of our results.

4.1.2 Vector Representation

In order to refer to a specific Euler-Lagrange equation or reference a particular element of the path q we require the ability to write specific vectors. The Euclidean space type we have used in chapter 3 for all of our vectors does not allow us to do this. We must therefore select an instance of the Euclidean space type to represent vectors in our system locales.

For one dimensional systems there exists a single obvious choice, we can use the type

`real`. It is proven in Isabelle that the real numbers are an instance of a Euclidean space where each of the vector specific operations are translated to real numbers in the obvious manner.

To represent vectors in an n dimensional system where n is greater than one we use the type `real^n`. With this type we use the `$` function to refer to individual elements of a given vector, for instance `q $ 2` refers to the second element of q. For the basis vectors we use `axis`, which represents a vector pointing along a single axis. This is a function of two variables, an index and a magnitude. The index specifies which of the axes the vector points along, while the magnitude takes the obvious meaning and is required to be 1 for the elements of the basis.

We initially found the type `real^n` difficult to understand, so in early versions of the project tuples of reals were used to represent vectors. A tuple of real numbers is essentially a row vector, and Isabelle allows us to treat it as such. This made writing both statements and their proofs relatively easy.

We do not continue to use the tuple representation because it results in less readable proofs, especially as the required dimension increases. One problem is that the tuple representation can consume a significant amount of space, which can make equations involving more than a single vector unreasonably long. Another problem is specifying a particular element of the vector. We could do this either through repeated applications of the the `fst` and `snd` functions, or using inner products with basis vectors. Neither one makes it easy to see which element of the vector is being referred to. While these problems do not make the representation unusable for low dimensional problems, in Lagrangian mechanics it is possible for complex systems to require a large number of dimensions to describe.

4.1.3 Equations

When we write a locale to represent a system, frequently one of our goals will be to derive the equations of motion for that system. It would be possible to derive each of these equations as a single lemma, however the resulting proofs would be very long and difficult to follow. We find it significantly improves readability to split the derivations into smaller parts.

We find it convenient to derive an equation for each of $\partial L/\partial q$, $\partial L/\partial \dot{q}$, and $\mathrm{d}(\partial L/\partial \dot{q}_i)/\mathrm{d}t$. The proofs for each equation are short enough to be easy to follow, and the terms themselves are meaningful so the result can be referenced without ambiguity. From these equations it is easy to state the Euler-Lagrange equation and derive an equation of motion.

When considering systems with more than one dimension we establish one set of equations for each dimension. It is frequently the case that the same term along different dimensions have very similar equations and proofs. For small dimensions, we find it easiest to consider these as separate equations and follow a similar proof structure. When considering problems of higher dimension it can be better to combine these proofs to a single lemma, although the resulting proof will be slightly more challenging.

Note that we do not establish an equation for q, but rather derive the differential equation q must satisfy. While in some cases it would be possible to take the derivation further, the steps to solving this differential equation for a general q or showing a specific form of q is a solution are no different in Lagrangian mechanics than in any other context. These steps would therefore be entirely independent of our mechanization of Lagrangian mechanics and so are not of interest to us for this project.

4.2 Example Systems

In this section we will discuss five systems which we have formalised in Isabelle which demonstrate the correctness and usability of our Lagrangian mechanics locale discussed in chapter 3. We start with the free particle system, where we have chosen to include the equations for the terms discussed in section 4.1.3 for illustrative purposes. The equations for these terms will usually not be discussed for brevity. We then describe harmonic oscillators, which are a commonly used system in physics. Projectile motion is our first system which required more than one dimension. The pendulum was interesting because it is the only system we have formalised which required q be thought of as an angle, while the Kepler problem was the most complex system for which we derived equations of motion. The central potential system will be discussed in section 5.2.3 and not here because we derived a conservation law rather than equations of motion for that system.

In the definition of each locale we make an assumption of the form `"lagrangian t q' qdot' = ..."`. The use of the prime notation distinguishes the parameters of the Lagrangian `q'` and `qdot'` from the functions `q` and `qdot` while not obscuring their relation. Importantly, the prime should not be confused for differentiation.

4.2.1 Free Particle

The free particle locale represents the system consisting of a single particle which is not acted on by any forces. Sussman uses this example as a very simple introductory problem with the following Lagrangian.

$$L(t,x,v) = \frac{1}{2}m(v \cdot v)$$

The system itself is not particularly interesting, but does place some restrictions on q. The simplicity of the system while maintaining the ability to derive some result made it an ideal initial system to represent using Lagrangian mechanics. While it would be reasonable to consider the system as having up to three dimension, we restrict the problem to a single dimension for even greater simplicity.

We start with the locale definition, which specifies the equation for the Lagrangian, fixes constants and states the required assumptions. Even in this very simple system there exist a constant which must be fixed, here mass represented by m. There is also

an assumption we must make, that $\dot{q}$ is a differentiable function. This is because the Euler-Lagrange equation results in taking its derivative.

```
locale free_particle = lagrangian_mechanics lagrangian q for
   lagrangian and q::"real ⇒ real" +
  assumes lagrangian_def: "lagrangian t q' qdot' = 1/2 * m *
  qdot'^2"
    and qdot_differentiable: "⋀t. qdot differentiable (at t)"
```

Since we are assuming that $\dot{q}$ is differentiable we choose to give its derivative a name. In physics the second derivative with respect to time is often written as $\ddot{q}$, so similar to our naming of $\dot{q}$ we choose the name `qdotdot` and define it as

```
definition qdotdot :: "real ⇒ real"
  where "qdotdot t ≡ (vector_derivative qdot) (at t)"
```

We then derive the intermediate equations discussed in section 4.1.3. Since we have restricted the locale to describe the one dimensional version of the system we have only three equations.

```
lemma q_partial_equation:
  shows "q_partial 1 t = 0"

lemma qdot_partial_equation:
  shows "qdot_partial 1 t = m * qdot t"

lemma dt_qdot_partial_equation:
  shows "dt_qdot_partial 1 t = m * qdotdot t"
```

Using these equations we can apply the Euler-Lagrange equation to derive the following equation of motion.

```
lemma equation_of_motion:
  shows "m * qdotdot t = 0"
```

In Newtonian mechanics one can recognise $m\ddot{q}$ for constant m as being equivalent to the net force by Newton's second law. The free particle is defined by the net force being equal to zero, so our equation of motion is precisely what we wished to derive.

We do not include a definition of force in any of our locales. Although force plays a very fundamental role in Newtonian mechanics, it is not significant in Lagrangian mechanics. Such a formalisation of force would also imply q represents linear motion, which will not always be the case. For instance, in the simple pendulum locale discussed in section 4.2.4 q represents an angle.

4.2.2 Harmonic Oscillator

A harmonic oscillator is a system with an equilibrium position and a restoring force proportional to the displacement from the equilibrium position. This description can apply to a variety of systems, so it may be simplest to think of a harmonic oscillator as anything which behaves like a spring.

Sussman states that a Lagrangian for a harmonic oscillator is

$$L(t,x,v) = \frac{1}{2}mv^2 - \frac{1}{2}kx^2$$

Similar to the free particle in section 4.2.1 the harmonic oscillator is described by a single dimension, however the harmonic oscillator is far more interesting because it can be used to describe many commonly occurring real world systems. We formalise this system in the following locale, which fixes a mass m and restoring force coefficient[1] k.

```
locale harmonic_oscillator = lagrangian_mechanics lagrangian q for
   lagrangian and q::"real ⇒ real" +
  assumes lagrangian_def: "lagrangian t q' qdot' = 1/2 * m * qdot'^2
    - 1/2 * k * q'^2"
    and qdot_differentiable: "⋀t. qdot differentiable (at t)"
```

Using the standard derivation we arrive at the following equation of motion.

```
lemma equation_of_motion:
  shows "m * qdotdot t = - k * q t"
```

This is equivalent to the equation Sussman derives, and is sometimes used to define a harmonic oscillator.

4.2.3 Projectile Motion

Projectile motion describes a particle moving horizontally under the influence of no forces and vertically under the force of gravity. This system is interesting as it requires two dimensions to describe, one for horizontal motion and one for vertical motion.

Sussman states the following Lagrangian for this system, where x represents horizontal motion and y vertical motion.

$$L\left(t, \begin{bmatrix} x \\ y \end{bmatrix}, \begin{bmatrix} v_x \\ v_y \end{bmatrix}\right) = \frac{1}{2}m(v_x^2 + v_y^2) - mgy$$

We formalise this system in the following locale, which fixes a mass m and gravitational acceleration g. We make the assumption that mass is positive so that we can derive the equation of motion in a familiar form. This assumption does not result in a loss of generality as a particle of zero mass[2] would not traditionally be thought of as following projectile motion. Note that our equation for the Lagrangian considers the first axis to be horizontal motion and the second axis to be vertical motion.

```
locale projectile_motion = lagrangian_mechanics lagrangian q for
   lagrangian and q::"real ⇒ real^2" +
  assumes lagrangian_def: "lagrangian t q' qdot' = (1/2 * m) *
  (qdot' • qdot') - m * g * q' $ 2"
    and positive_mass: "m > 0"
    and qdot_differentiable: "⋀t. qdot differentiable (at t)"
```

[1] When the harmonic oscillator is a spring this is commonly referred to as the spring constant.

[2] Or negative mass, if you would like to consider that possibility.

We then derive the intermediate equations. Since this is a two dimensional system, there will be two equations for each term mentioned in section 4.1.3. For the sake of brevity we list only the two equations for $\partial L/\partial \dot{q}$, though comparable pairs of equations exist for the other two terms.

```
lemma qdot_partial_axis_1:
  shows "qdot_partial (axis 1 1) t = m * qdot t $ 1"
```

```
lemma qdot_partial_axis_2:
  shows "qdot_partial (axis 2 1) t = m * qdot t $ 2"
```

You may note these two equations are very similar, and in fact their proofs are nearly identical. As discussed in section 4.1.3 it would be possible to combine these two equations into a singe lemma, however we do not find this beneficial for only two dimensions.

We ultimately derive two equations of motion, one for each axis, using the Euler-Lagrange equations.

```
lemma equation_1:
  shows "qdotdot t $ 1 = 0"
```

```
lemma equation_2:
  shows "qdotdot t $ 2 = -g"
```

These equations say that there is no acceleration in the horizontal direction, while the acceleration in the vertical direction is negative g. The negative sign comes from the acceleration being downwards, with up being considered positive. These are the same results Sussman derives.

4.2.4 Simple Pendulum

In this section we consider a pendulum of fixed length subject to no external forces except gravity. It is possible to describe the position of such a pendulum by the angle it makes with the vertical line through its anchor point. Sussman describes such a pendulum using the following Lagrangian.

$$L(t,\theta,\dot{\theta}) = \frac{1}{2}ml^2\dot{\theta}^2 + mgl\cos\theta$$

We formalise this system in the following locale, which fixes a mass m, pendulum length l, and gravitational acceleration g. We assume that both the mass and length are positive so that we can more easily compare our equation of motion with the equation typically used in physics. These assumptions are justified, a pendulum of zero mass cannot be acted on by gravity and a pendulum of zero length is not a pendulum.

```
locale simple_pendulum = lagrangian_mechanics lagrangian q for
   lagrangian and q::"real ⇒ real" +
  assumes lagrangian_def: "lagrangian t q' qdot' =  1/2 * m * l^2 *
   qdot'^2 + m * g * l * cos q'"
    and positive_mass: "m > 0"
    and positive_length: "l > 0"
    and qdot_differentiable: "⋀t. qdot differentiable (at t)"
```

We find this system interesting as it is the first of our formalisations where q is explicitly interpreted as an angle rather than a position in space. It is also the first of our formalisations in which trigonometric functions appear. These facts do not significantly change how we derive the following equation of motion.

```
lemma equation_of_motion:
  shows "l * qdotdot t + g * sin (q t) = 0"
```

The equation we have derived is an accurate description of the motion of a pendulum. It is interesting to note this is not the same equation most commonly used in physics, where the approximation $\sin\theta \approx \theta$ is used so that the pendulum can be described as a harmonic oscillator (see section 4.2.2) with $k = g/l$.

4.2.5 Kepler Problem

Sussman [22] describes the Kepler problem as

> the two-dimensional motion of a particle of mass m orbiting a fixed center of attraction, with gravitational potential energy $-\mu/r$, where r is the distance to the center of attraction.

They proceed to define the following Lagrangian for this system.

$$L\left(t, \begin{bmatrix}\xi\\ \eta\end{bmatrix}, \begin{bmatrix}v_\xi\\ v_\eta\end{bmatrix}\right) = \frac{1}{2}m(v_\xi^2 + v_\eta^2) + \frac{\mu}{\sqrt{\xi^2+\eta^2}}$$

We formalised this system in the following locale, which fixes a mass m and potential energy coefficient μ. We assumed that $q \bullet q$ is always positive, as if it were ever zero the Lagrangian would be undefined due to division by zero. This inner product is never negative because q is a vector of real numbers.

```
locale kepler_problem =  lagrangian_mechanics lagrangian q for
   lagrangian and q::"real ⇒ real^2" +
  assumes lagrangian_def: "lagrangian t q' qdot' =  1/2 * m *
  (qdot' • qdot') + mu / sqrt (q' • q')"
    and qdot_differentiable: "⋀t. qdot differentiable (at t)"
    and nonzero_path: "⋀t. q t • q t > 0"
```

We found this system interesting because it has a complicated Lagrangian involving both division and square roots. This made it it difficult to derive the equations of motion since the derivatives required several applications of the chain rule. We managed to derive the following equations of motion, which are equivalent to those derived by Sussman.

```
lemma equation_1:
  shows "m * qdotdot t $ 1 + mu * q t $ 1 /(sqrt (q t • q t))^3 = 0"
```

```
lemma equation_2:
  shows "m * qdotdot t $ 2 + mu * q t $ 2 /(sqrt (q t • q t))^3 = 0"
```

Chapter 5

Conserved Quantities

In this chapter we will discuss two conservation laws in Lagrangian mechanics. The first is the conservation of momentum, an intuitive result relating the independence of the Lagrangian from a coordinate q_i to a conserved quantity. The second is Noether's theorem. This is a much more involved conservation law, requiring a coordinate transform to be fixed and satisfy certain properties before determining the conserved quantity.

Before we can prove these conservation laws we need to define what we mean for a quantity to be conserved. The intuitive meaning is that the quantity does not change with time. In the language of calculus this idea is represented by the time derivative of the quantity equalling 0 at every point in time. We express this in Isabelle using the following definition, added to the Lagrangian mechanics definitions locale.

```
definition conserved :: "(real ⇒ 'b::euclidean_space) ⇒ bool"
  where "conserved f ≡ ∀t. (vector_derivative f) (at t) = 0"
```

You may notice that `conserved` applies to functions of the type $\mathbb{R} \to \beta$, where β is an arbitrary `euclidean_space`, rather than $\mathbb{R} \to \alpha$ or $\mathbb{R} \to \mathbb{R}$. We do this because we may want to state that a function of either type is a conserved quantity, but do not want to write separate definitions for the two.

There are other points in this chapter where we shall use β to represent an arbitrary `euclidean_space`. It should be noted that these instances of β do not need to represent the same `euclidean_space`. This differs from the behaviour of α, which must always represent the same `euclidean_space`, because α is fixed within the context of the locale while β is not.

5.1 Conservation of Momentum

The momenta conservation law states that if the Lagrangian does not depend on an element of q, say the ith element, then $\partial L/\partial \dot{q}_i$ is a conserved quantity. The proof is quite simple. Since the Lagrangian does not depend on q_i we have $\partial L/\partial q_i = 0$, so the ith Euler-Lagrange equation is

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{q}_i} = 0.$$

Using our definition of conserved, we see this is precisely the claimed result. We implement this proof in the following Isabelle Lemma.

```
lemma momenta_conservation_law:
  assumes "v∈Basis"
  assumes "∀t. q_partial v t = 0"
  shows "conserved (qdot_partial v)"
```

To make sure the result we have derived can be applied, we will use it to derive a conservation law for the free particle locale discussed in section 4.2.1. Recall that the Lagrangian for this system did not depend on q. In fact, we have already proven this while deriving the equation of motion. Using this we prove the following conservation law in the locale.

```
lemma conserved_momenta:
  shows "conserved (λt. m * qdot t)"
```

5.2 Noether's Theorem

Noether's theorem is a very powerful and beautiful result which relates continuous symmetries with conserved quantities of a system. It is also challenging to prove. In this section we will first discuss preliminary work required to prove Noether's theorem, specifically a proof of a particular form of the chain rule. We will then prove Noether's theorem, and discuss the assumptions we require. Finally we will discuss a new locale where we will use Noether's theorem to derive a conservation law, showing our proof is both correct and usable.

5.2.1 Multiple variable chain rule

A major step in proving Noether's theorem is an application of the chain rule. In Isabelle there exist multiple proofs of the chain rule for differing definitions of the derivative, you may recall we have already used one of these definitions in section 4.2.5. The version of the chain rule we require takes the form

$$\frac{\mathrm{d}}{\mathrm{d}t} f \begin{pmatrix} x_1(t) \\ \vdots \\ x_n(t) \end{pmatrix} = \frac{\partial f}{\partial x_1}\frac{\mathrm{d}x_1}{\mathrm{d}t} + \cdots + \frac{\partial f}{\partial x_n}\frac{\mathrm{d}x_n}{\mathrm{d}t}. \tag{5.1}$$

This version of the chain rule very clearly depends on the partial derivative, which is not defined in Isabelle's default libraries, so the proof of this form of the chain rule cannot be found in Isabelle's multiple variable analysis library. It is also not included in Abdulaziz and Paulson's proof of Green's Theorem [1] in which the definition of partial derivatives we are using originates.

Before proving the chain rule itself, we find it helpful to derive the following theorem which allows us to write a function f of type $\beta \to \mathbb{R}$ as a sum of its partial derivatives multiplied by the basis vectors.

```
lemma derivative_to_sum:
  assumes "f differentiable (at x)"
  shows "(f has_derivative (λu. u • (Σv∈Basis.
    partial_vector_derivative f v x *R v))) (at x)"
```

Up until this point we have avoided using Isabelle's definition `has_derivative`, favoring `vector_derivative` and `partial_vector_derivative` for reasons discussed in section 3.2. We make an exception for this lemma and its applications because we are required to use `has_derivative`. It is impossible to use `vector_derivative` because the type we have specified for f does not match the type of functions for which `vector_derivative` is defined.

This proof was tricky because it required a detailed understanding of the definitions of multiple forms of the derivative and the relations between those definitions.

It was also quite challenging on a conceptual level because it uses the Fréchet derivative. We have no prior experience with this form of the derivative, and identified in section 3.2 that its definition does not allow us to substitute results we know to hold for ordinary, familiar forms of the derivative without some changes. This lack of experience meant we had little intuition for what statements we could expect to be true, and even less about what steps needed to be taken to prove them.

To prove the chain rule we introduce a second function g which is of type $\mathbb{R} \to \beta$. The composition $f \circ g$ will then be a function of type $\mathbb{R} \to \mathbb{R}$, just as in equation 5.1.

```
lemma multiple_variable_chain_rule:
  assumes "(g has_vector_derivative g' x) (at x)"
  assumes "f differentiable (at (g x))"
  shows "vector_derivative (f ∘ g) (at x) = (Σv∈Basis.
  (partial_vector_derivative f v (g x)) * (v • g' x))"
```

A major step in this proof was using one of the definitions of the chain rule proven in Isabelle combined with our earlier lemma. This allowed us to write the derivative as the inner product of two vectors of type β, from which the result follows by applying properties of euclidean spaces.

Comparing our proven lemma with equation 5.1 it would appear that we are done, however this is not the case. Because of the way we have defined the Lagrangian, to prove Noether's theorem we will actually require a version of the chain rule which applies to a function of two vectors.

Before we can do this we will need a pair of supporting lemmas. These lemmas include Isabelle set comprehension, which is defined very similarly to list comprehension in languages like Python and Haskell. The set `{(v, a) |v. v∈Basis)}` are all tuples of the form `(v, a)` such that `v∈Basis`.

Our first lemma states that the sum of the values of any function on the set `Basis` is equal to the sum of the values of that function on the set `{(v, a) |v. v∈Basis)}` if

we first discard the second element of the tuple. In this lemma the second element `a` is a completely arbitrary value of arbitrary type.

```
lemma tupled_sum:
  shows "(Σv∈Basis. f v) = (Σv∈{(v, a) |v. v∈Basis)}. f (fst v))"
```

The second makes an equivalent statement to the first, but switches the order of the elements in the tuple.

```
lemma tupled_sum2:
  shows "(Σv∈Basis. f v) = (Σv∈{(a, v) |v. v∈Basis}. f (snd v))"
```

We prove the first lemma using induction on the elements of `Basis`, while the second lemma follows from the first using a function which flips the order of elements in the tuple.

We now formalise our second variation of the chain rule in the following lemma, f now has type $\beta \to \beta \to \mathbb{R}$ and in addition to g we have introduce a second function h also of type $\mathbb{R} \to \beta$. This makes our differentiability assumption on f more complicated, but otherwise does not significantly change the statement.

```
lemma multiple_variable_chain_rule2:
  assumes "(g has_vector_derivative g' x) (at x)"
  assumes "(h has_vector_derivative h' x) (at x)"
  assumes "(λv. f (fst v) (snd v)) differentiable at (g x, h x)"
  shows "vector_derivative (λy. f (g y) (h y)) (at x) = (Σv∈Basis.
  (partial_vector_derivative (λy. f y (h x)) v (g x)) * (v • g' x) +
  (partial_vector_derivative (λy. f (g x) y) v (h x)) * (v • h' x))"
```

Intuitively this is a reasonably obvious result, instead of a function of two vectors we could define an equivalent function of a single vector which is twice[1] the length. We follow similar reasoning in our Isabelle proof. Using tuples to combine the vectors in a single vector, we define two new functions F and G such that $F \circ G(y) = f(g(y), h(y))$.

```
  let ?F = "λv. f (fst v) (snd v)"
  let ?G = "λy. (g y, h y)"
```

The functions F and G satisfy the assumptions of our first proof of the chain rule. We apply this to express the derivative of $F \circ G$ as a sum over a basis. It is important to note that the basis that appears in the result of applying the chain rule to $F \circ G$ is a set of tuples rather than the basis associated with the type β, while the result we are trying to prove includes a sum over β's basis.

We define two new sets, B_1 and B_2, which will partition the tuple basis. B_1 will contain all the elements of the tuple basis where the first element of the tuple is an element of β's basis, while B_2 does the same but for the second element.

```
  let ?B1 = "{(v,0) |v. v∈Basis}"
  let ?B2 = "{(0,v) |v. v∈Basis}"
```

[1] More generally the vectors would not be required to be the same length, in which case the new function would be of a single vector which is the sum of the lengths of the two vectors. This level of generality is not required for our application of the chain rule.

By splitting the sum which appears from applying the chain rule to $F \circ G$ into a sum over B_1 plus a sum over B_2, through some manipulation of the definition of the partial derivative we get expressions to which we can apply our supporting lemmas. The remaining steps of the proof are algebraic.

5.2.2 Proving Noether's Theorem

Noether's theorem states that for every continuous symmetry of the Lagrangian there exists an associated quantity which is conserved. A symmetry is a property which is unchanged under some transformation. In Noether's theorem, the unchanged property is the Lagrangian and the transformation is a change to q[2]. We must formalise both the change to q and the fact that this does not change the Lagrangian to prove Noether's theorem.[3]

Throughout this section we will use transformed Lagrangian to refer to the Lagrangian in which the change to q has been made, while original Lagrangian will refer to the Lagrangian without this change.

Sussman's proof of Noether's theorem first defines a parametric coordinate transform function. The change to q was formalised as an application of this coordinate transform. That this coordinate transform did not change the Lagrangian was formalised as an equality between the transformed Lagrangian and original Lagrangian. One issue we had was interpreting Sussman's notation, which made the proof confusing as it did not clearly distinguish between the coordinate transform on its own and the coordinate transform applied to q.

David Tong [23] proves Noether's theorem very similarly to Sussman, but formulates the requirements differently. Instead of defining a coordinate transform function then apply it to q, Tong creates a single definition for a transformed path function. This prevented any ambiguity as to whether a property of the coordinate transform or the coordinate transform applied to q was being discussed. Tong formalised that this transformed path did change the Lagrangian as the derivative of the transformed Lagrangian with respect to the parameter equalling zero.

The following is our formalisation of Noether's theorem, where Q is a transformed path function which has type $\mathbb{R} \to \mathbb{R} \to \alpha$. The first argument of Q is a parameter, which we will denote by s, and the second time.

```
lemma noether:
  assumes equal_at_0: "Q 0 = q"
  assumes invariant_wrt_s: "⋀s t. lagrangian t (q t) (qdot t) =
    lagrangian t (Q s t) ((vector_derivative (Q s)) (at t))"
  assumes Q_deriv_param: "⋀t. (λs. Q s t) differentiable (at 0)"
  assumes Q_deriv_time: "⋀t. (λt'. Q s t') differentiable (at t)"
  assumes Q_second_deriv: "⋀t. (λs. (vector_derivative
  (λt'. Q s t')) (at t)) differentiable (at 0)"
  assumes reorder_derivs: "⋀t. sym_deriv Q 0 t"
```

[2] This will also change $\dot{q}$ to be the time derivative of the changed q.

[3] It would be nice to define an Isabelle predicate stating a particular function is a symmetry. Unfortunately, we find the general definitions in literature too vague to formalise as a definition.

```
  shows "conserved (λt. qdot_partial_vector t • ((vector_derivative
    (λs. Q s t)) (at 0)))"
```

Our statement of Noether's theorem is not the same as either Sussman's or Tong's, instead choosing to include the aspects of each we believe to be the most clear and elegant. We also considered how easy it would be to show the requirements of Noether's theorem are satisfied when we use it to derive a conservation law. We will see in section 5.2.3 that our version is still an accurate statement of Noether's theorem by applying it to an example provided by Sussman and showing we derive the same result.

Our first decision was to use Tong's transformed path notation rather than Sussman's coordinate transform. We find this notation easier to follow and more closely matches our intuition, which makes the resulting proof more readable. Using the coordinate transform notation would also likely require using additional parentheses to make our intended meaning clear.[4] We already use a large number of parentheses in each line of our proof, too many results in a less readable proof.

Our first assumption states that $Q(0,t) = q(t)$. This is identical to an assumption in Tong's proof, and an equivalent statement is made by Sussman using the coordinate transform notation.

Our second assumption is that the transformed Lagrangian is equal to the original Lagrangian for all values of time t and parameter s. This is very similar to the assumption made in Sussman's proof. Tong assumption, the derivative with respect to s of the transformed Lagrangian is 0, is derived as an early step in our proof. We chose this form of the assumption because we felt it more fundamentally represents the idea of a symmetry.

The remaining assumptions all relate to the differentiability of Q. The last of these assumptions is quite interesting. It states that when we take both a time derivative and a parameter derivative of Q, changing the order in which we take these derivatives does not change the value. We have formalised this in Isabelle using the following abbreviation.

```
abbreviation "sym_deriv f s t ≡ ∀g. ((λs'. ((vector_derivative (λt'.
    f s' t')) (at t))) has_vector_derivative g) (at s) ⟷ ((λt'. ((
   vector_derivative (λs'. f s' t')) (at s))) has_vector_derivative
   g) (at t)"
```

The equation requiring this assumption is difficult to spot and entirely unexplained in Tong's proof. Sussman on the other hand points out this requirement in a footnote, alongside an explanation for why it is satisfied. We found this explanation lacked the detail required to convince us Sussman's claim is correct, and was certainly insufficient for us to attempt to prove it.

It is not generally true that derivatives are commutative. We can show this using a counterexample, such as the function

[4] While Isabelle has tools for specifying the default order of operations which would technically mean we don't require the additional parentheses, the resulting proof would likely not be as human readable as we desire.

$$f(x,y) = \begin{cases} \frac{xy(x^2-y^2)}{x^2+y^2} & \text{if} \quad (x,y) \neq (0,0) \\ 0 & \text{if} \quad (x,y) = (0,0) \end{cases}$$

If we differentiate this function first with respect to x then with respect to y, then we find the second derivative at the origin is equal to -1. If we were to reverse the order in which we take the derivatives we find second derivative at the origin is equal to 1.

It is possible to show that derivatives are commutative under certain conditions. For instance, Clairaut's theorem states that if both orderings of the partial derivatives exist and are continuous in a region about a point then they are equal at that point. To use such results to prove Noether's theorem we would need to make strong assumptions, which results in a less general statement of the theorem. This is undesirable.

Proving conditions under which the derivatives are commutative could be a useful expansion to this project. While not useful when proving Noether's theorem, these conditions could be used to discharge assumptions when attempting to apply Noether's theorem. For instance, Clairaut's theorem implies many commonly used functions, such as the polynomial, exponential, sin, and cos functions, are twice differentiable with equal mixed partial derivatives. This could then be extended to sums, products, and compositions of such functions using rules of the derivative and continuity.

5.2.3 Central Potential locale

Having proved Noether's theorem, we would like to apply it to an example system. This is important for ensuring the assumptions of our proof are satisfiable and that they are not significantly stricter than required, which would make the theorem difficult to apply. Sussman provides such an example, applying Noether's theorem to a system described by the Lagrangian

$$L\left(t, \begin{pmatrix} x \\ y \\ z \end{pmatrix}, \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix}\right) = \frac{1}{2}m(v_x^2 + v_y^2 + v_z^2) - U(\sqrt{x^2+y^2+z^2}).$$

This system is fairly interesting. The Lagrangian includes a function U which we know essentially[5] nothing about. If we were to attempt to apply the Euler-Lagrange equations to this system our result would depend on the derivative of U, so would not provide significant information regarding the motion of the system. This makes it somewhat surprising that Noether's theorem will allow us to derive a conservation law for this system.

We implement this system in Isabelle as the central potential locale. Just as in Sussman's definition U is a function of type $\mathbb{R} \to \mathbb{R}$ while m is a constant of type $\mathbb{R}$. Note that we have replaced Sussman's sums of squares of components with an inner product. This makes our representation more compact for better readability.

[5]From assumptions made about the Lagrangian there is some information we can derive about U. For instance, since the Lagrangian is assumed to be differentiable we can infer that U must also be differentiable.

```
locale central_potential = lagrangian_mechanics lagrangian q for
   lagrangian and q::"real ⇒ real^3" +
  assumes lagrangian_def: "lagrangian t q' qdot' =  1/2 * m * (qdot'
    • qdot') - U (sqrt (q' • q'))"
```

Sussman defines a parametric rotation about the z-axis as

$$R_z(s)\begin{pmatrix}x'\\y'\\z'\end{pmatrix} = \begin{pmatrix}x'\cos s - y'\sin s\\ x'\sin s + y'\cos s\\ z'\end{pmatrix}$$

Using R_z and Noether's theorem they prove that $m(yv_x - xv_y)$ is a conserved quantity.

We will prove the same result in our Isabelle locale. Our first step is to define the rotation about the z-axis in Isabelle. We do this by expressing R_z as the elements of R_z multiplied by the corresponding basis vectors.

$$R_z(s)\begin{pmatrix}x'\\y'\\z'\end{pmatrix} = (x'\cos s - y'\sin s)\begin{pmatrix}1\\0\\0\end{pmatrix} + (x'\sin s + y')\begin{pmatrix}0\\1\\0\end{pmatrix} + z'\begin{pmatrix}0\\0\\1\end{pmatrix}$$

This becomes the following Isabelle definition.

```
definition z_rotation :: "real ⇒ real^3 ⇒ real^3"
  where "z_rotation s x ≡
            (x $ 1 * cos s - x $ 2 * sin s) *R (axis 1 1)
          + (x $ 1 * sin s + x $ 2 * cos s) *R (axis 2 1)
          + (x $ 3) *R (axis 3 1)"
```

Defining R_z as a sum of functions multiplied by basis vectors feels less elegant that defining it to be a single vector, the elements of which are functions. Unfortunately we believe our less elegant presentation is the best we can do while using Isabelle's `vec` type to represent vectors since the type does not offer a convenient method for writing a vector explicitly. It would be possible to represent R_z more elegantly if we chose to represent vectors using tuples rather than `vec`. We choose not to take this approach to maintain consistency with earlier locales in addition to the reasons discussed in section 4.1.2.

Using our understanding of the sin and cos functions it is apparent that $R_z(0)$ is the identity function. Sussman considers this step obvious enough to omit it from his example. Isabelle does not allow such omissions, so we provide a proof of the statement.

```
lemma z_rotation_at_zero:
  shows "z_rotation 0 x = x"
```

This would discharge the first assumption of Noether's theorem had we used Sussman's formulation. In our version of Noether's theorem we require the transformed path function. This is easy to derive from the coordinate transform, which we do with the following definition.

```
definition z_rotation_path :: "real ⇒ real ⇒ real^3"
  where "z_rotation_path s t ≡ z_rotation s (q t)"
```

When we apply Noether's theorem we will specify that the role of Q is played by `z_rotation_path`. We must therefore show every assumption of Noether's theorem is satisfied for `z_rotation_path`. From proving `z_rotation_at_zero` we can easily derive the following lemma, which discharges the first assumption of Noether's theorem.

```
lemma z_rotation_path_at_zero:
  shows "z_rotation_path 0 = q"
```

The next assumption we need to discharge is that the transformed Lagrangian is equal to the original Lagrangian. To do this we must first find the time derivative of `z_rotation_path` since the transformed Lagrangian will depend on this quantity. It is clear to us this will be $R_z(s)$ applied to $\dot{q}$, which we prove in the following lemma.

```
lemma z_rotation_path_time_derivative:
  shows "((z_rotation_path s) has_vector_derivative z_rotation s (
    qdot t)) (at t)"
```

We know a rotation about an axis is an orthogonal linear transform, which by definition means the inner product of any pair of vectors is unchanged when the rotation is applied to both vectors. Since `z_rotation` appears both in our definition `z_rotation_path` and the form we have just derived to its time derivative, it will be helpful to show this property. From the form of the Lagrangian we observe that we only require the case where the inner product is between two equal vectors. We prove this in the following lemma.

```
lemma z_rotation_inner:
  shows "z_rotation s v • z_rotation s v = v • v"
```

From this lemma we prove that the transformed Lagrangian is equal to the original Lagrangian. This discharges the second assumption of Noether's theorem.

```
lemma z_rotation_unchanged_lagrangian:
  shows "lagrangian t (q t) (qdot t) = lagrangian t (z_rotation_path
    s t) (vector_derivative (z_rotation_path s) (at t))"
```

From the two properties we have shown thus far, most proofs would consider the requirement Noether's theorem to be fulfilled. We are not quite done as our formalisation of Noether's theorem, found in section 5.2.2, required several assumptions regarding the derivatives of Q.

We must show that both first derivatives of `z_rotation_path` exist. We have already found its time derivative as a intermediate step to discharging earlier assumptions, which shows it is differentiable with respect to time. We now define a function which is the derivative with respect to the parameter, then prove it is the derivative we claim.

```
definition ds_z_rotation_path :: "real ⇒ real ⇒ real^3"
  where "ds_z_rotation_path s t ≡ (- q t $ 1 * sin s - q t $ 2 * cos
    s) *R axis 1 1 + (q t $ 1 * cos s - q t $ 2 * sin s) *R axis 2 1"
```

```
lemma z_rotation_path_parameter_derivative:
  shows "((λs. z_rotation_path s t) has_vector_derivative
    ds_z_rotation_path s t) (at s)"
```

This will discharge an assumption of Noether's theorem. The result we have proven is actually stronger than required, since it shows `z_rotation_path` is differentiable for every value of s. The assumption in Noether's theorem only requires that the derivative exist at $s = 0$.

There are two final assumptions we need to discharge, both relating to the second derivatives of `z_rotation_path`. Taken together they mean both mixed derivatives exist and the order in which the derivatives are taken does not change the value of the derivative. We show these properties by calculating both orders of the derivative and observing they take the same value.

```
lemma z_rotation_path_time_parameter_derivative:
  shows "((λs. vector_derivative (z_rotation_path s) (at t))
    has_vector_derivative (- qdot t $ 1 * sin s - qdot t $ 2 * cos s)
     *R axis 1 1 + (qdot t $ 1 * cos s - qdot t $ 2 * sin s) *R axis 2
    1) (at s)"
```

```
lemma z_rotation_path_parameter_time_derivative:
  shows "((λt. vector_derivative (λs. z_rotation_path s t) (at s))
    has_vector_derivative (- qdot t $ 1 * sin s - qdot t $ 2 * cos s)
     *R axis 1 1 + (qdot t $ 1 * cos s - qdot t $ 2 * sin s) *R axis 2
    1) (at t)"
```

The final lemma we shall prove before applying Noether's theorem is an equation for $\partial L/\partial \dot{q}_i$. Our conclusion to Noether's theorem states the conservation law as an inner product. It is not possible to look at this inner product and see precisely what quantity is conserved without some effort and knowing what each vector represents. We will use this lemma to rewrite the conserved quantity in a more readable form, which will help us compare our results with Sussman's.

```
lemma qdot_partial_axis:
  assumes "v∈Basis"
  shows "qdot_partial v t = m * (qdot t • v)"
```

We have now proven all assumptions of Noether's theorem are satisfied when using `z_rotation_path` for the transformed path and derived an additional lemma to help us re-express the conclusion of Noether's theorem in a manner which makes the conserved quantity obvious. Putting these pieces together, we prove the following conservation law.

```
lemma z_rotation_conservation_law:
  shows "conserved (λt. m * ((q t $ 1) * (qdot t $ 2) -  (q t $ 2) *
     (qdot t $ 1)))"
proof -
  ...
  ultimately have "conserved (λt. qdot_partial_vector t •
    vector_derivative (λs. z_rotation_path s t) (at 0))"
     using noether [where Q=z_rotation_path] by blast
  moreover have "(λt. qdot_partial_vector t • vector_derivative (λs.
     z_rotation_path s t) (at 0)) = (λt. m * ((q t $ 1) * (qdot t $
    2) - (q t $ 2) * (qdot t $ 1)))"
     <proof>
  ultimately show ?thesis
     by simp
```

The conservation law we have derived is equivalent to the one derived by Sussman[6], which provides confidence that our conclusion to Noether's theorem is correct. No additional assumptions beyond those specified by the `lagrangian_mechanics` locale were required, indicating our conditions for applying Noether's theorem are likely no more restrictive than necessary. We already believed both of these things to be true as we spend a great deal of time considering how best to formalise Noether's theorem in Isabelle. It is nonetheless nice to have an example which supports our belief.

Proving this conservation law has also helped us understand how challenging it can be to show the assumptions of Noether's theorem are satisfied. Compared to the other locales representing a physical system, discussed in section 4.2, the central potential locale is significantly longer and contains more complex proofs. We attributed this complexity entirely to Noether's theorem as every proof in this locale was working towards a single conservation law.

We believe it may be possible to prove supporting theorems which would make it easier to apply Noether's theorem in many cases. For instance, at the end of section 5.2.2 we discussed how Clairaut's theorem could be used to discharge the assumptions regarding derivatives of Q. In this locale we discharged these assumptions by calculating all four relevant derivatives. These are not the most complicated proofs in this locale, but neither are they trivial. A simpler and faster method would have been preferred.

It would also be possible to prove other methods for deriving conservation laws with assumptions which are easier to show. We have proven the momenta conservation law, discussed in section 5.1, which is one such method. Our example of using this method to show a quantity is conserved could also have used Noether's theorem with $Q = q + s$. This would have required several steps to show the assumptions of Noether's theorem were satisfied, while the momenta conservation law was trivial to apply.

[6] We show $m(xv_y - yv_x)$ is conserved while Sussman shows $m(yv_x - xv_y)$ is conserved. These appear to be different, however one can be obtained from the other by multiplying by -1. It is clear from the definition of conservation that a conserved quantity multiplied by a constant is still a conserved quantity.

Chapter 6

Conclusion

We have established a locale formalising Lagrangian mechanics, where we have treated the Euler-Lagrange equations as first principles rather than the principle of least action. The challenges in this section of the project where primarily understanding derivatives in Isabelle well enough to write the required assumptions.

The derivatives again proved challenging when we started to write the locales describing specific systems, described in section 4.2. This was unexpected since we had extensive experience with derivatives in pen and paper mathematics, yet were not able to prove very basic derivatives. We discovered that many of the theorems needed us to use `where [...]` to specify the value for several terms in the theorem for Isabelle to figure out our proof was correct. Once we figured this out the proofs became somewhat formulaic. We attempted to choose interesting systems to counterbalance this.

The derivation of Noether's theorem was quite difficult, which we had expected. The application of Noether's theorem being as long and complicated as it turned out to be came as a surprise. Sussman's application of Noether's theorem was very brief and contained steps which seemed obvious, yet did not turn out to be so simple in Isabelle.

One possible expansion for this project was discussed in section 5.2.3, consisting of proving results which would make Noether's theorem easier to apply or proving alternative conservation laws. Another possible expansion which would not diverge too far from the aims of this project would be to derive the equations of motion for a system described by special relativity rather than classical mechanics. We believe this would not be significantly different from the systems described in section 4.2, except the Lagrangian for the system would likely be more complicated.

Another, more advanced expansion would be to use our formalisation of Lagrangian mechanics as the basis for a formalisation of Hamiltonian mechanics. We are not sure how challenging this task would be as we have no experience with Hamiltonian mechanics, but know it is closely related to Lagrangian mechanics.

Bibliography

[1] Mohammad Abdulaziz and Lawrence C. Paulson. An Isabelle/HOL Formalisation of Green's Theorem. *Journal of Automated Reasoning*, 63:763–786, November 2018.

[2] Tom Mike Apostol. *Mathematical Analysis. 2nd ed.* Addison-Wesley, 1974.

[3] Katherine Brading, Elena Castellani, and Nicholas Teh. Symmetry and Symmetry Breaking. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, Fall 2021 edition, 2021.

[4] Binyameen Farooq, Osman Hasan, and Sohail Iqbal. Formal Kinematic Analysis of the Two-Link Planar Manipulator. In *International Conference on Formal Engineering Methods*, pages 347–362. Springer, 2013.

[5] Richard Feynman. Lectures on physics. https://www.feynmanlectures.caltech.edu/index.html.

[6] Alexander Gorban and Ilya Karlin. Hilbert's 6th problem: exact and approximate hydrodynamic manifolds for kinetic equations. *Bulletin of the American Mathematical Society*, 51(2):187–246, 2014.

[7] Yong Guan, Jingzhi Zhang, Guohui Wang, Ximeng Li, Zhiping Shi, and Yongdong Li. Formalization of Euler–Lagrange Equation Set Based on Variational Calculus in Hol Light. *Journal of Automated Reasoning*, 65:1–29, March 2020.

[8] David Hilbert. Mathematical problems. *Bulletin of the American Mathematical Society*, 8(10):437–479, 1902.

[9] Johannes Hölzl, Fabian Immler, and Brian Huffman. Type Classes and Filters for Mathematical Analysis in Isabelle/HOL. In Sandrine Blazy, Christine Paulin-Mohring, and David Pichardie, editors, *Interactive theorem proving 4th International Conference*, page 279–294. Springer Berlin Heidelberg, 2013.

[10] Omar A Jasim and Sandor M Veres. Verification Framework for Control System Functionality of Unmanned Aerial Vehicles. *arXiv preprint arXiv:2006.10860*, 2020.

[11] Eric Hanqing Lu. *A Formalization of Elements of Special Relativity in Coq*. PhD thesis, Harvard College, 2017.

[12] Sha Ma, Zhiping Shi, Zhenzhou Shao, Yong Guan, Liming Li, and Yongdong Li. Higher-Order Logic Formalization of Conformal Geometric Algebra and its Application in Verifying a Robotic Manipulation Algorithm. *Advances in Applied Clifford Algebras*, 26(4):1305–1330, 2016.

[13] John CC McKinsey, AC Sugar, and Patrick Suppes. Axiomatic Foundations of Classical Particle Mechanics. *Journal of rational mechanics and analysis*, 2:253–272, 1953.

[14] David Morin. *Introduction to Classical Mechanics: With Problems and Solutions*. Cambridge University Press, 2008.

[15] Isaac Newton. *Philosophiae Naturalis Principia Mathematica*, volume 1. G. Brookman, 1833.

[16] Richard Schmoetten, Jake E Palmer, and Jacques D Fleuriot. Towards Formalising Schutz'Axioms for Minkowski Spacetime in Isabelle/HOL. *arXiv preprint arXiv:2108.10868*, 2021.

[17] Zhiping Shi, Aixuan Wu, Xiumei Yang, Yong Guan, Yongdong Li, and Xiaoyu Song. Formal analysis of the kinematic Jacobian in screw theory. *Formal Aspects of Computing*, 30(6):739–757, 2018.

[18] Herbert A Simon. The Axiomatization of Classical Mechanics. *Philosophy of Science*, 21(4):340–343, 1954.

[19] Mike Stannett and István Németi. Using Isabelle/HOL to verify first-order relativity theory. *Journal of automated reasoning*, 52(4):361–378, 2014.

[20] James Stewart. *Calculus : Early Transcendentals*. Media Production Services Unit, Manitoba Education, 2012.

[21] Patrick Suppes. The Desirability of Formalization in Science. *The Journal of Philosophy*, 65(20):651–664, 1968.

[22] Gerald Jay Sussman and Jack Wisdom. *Structure and Interpretation of Classical Mechanics*. The MIT Press, 2015.

[23] David Tong. Lectures on Classical Dynamics. http://www.damtp.cam.ac.uk/user/tong/dynamics.html, January 2015.