Formalization of Transform Methods using HOL Light

Adnan Rashid and Osman Hasan
School of Electrical Engineering and Computer Science (SEECS),
National University of Sciences and Technology (NUST), Islamabad, Pakistan
{adnan.rashid,osman.hasan}@seecs.nust.edu.pk

Abstract

Transform methods, like Laplace and Fourier, are frequently used for analyzing the dynamical behaviour of engineering and physical systems, based on their transfer function, and frequency response or the solutions of their corresponding differential equations. In this report, we present an ongoing project, which focuses on the higher-order logic formalization of transform methods using HOL Light theorem prover. In particular, we present the motivation of the formalization, which is followed by the related work. Next, we present the task completed so far while highlighting some of the challenges faced during the formalization. Finally, we present a roadmap to achieve our objectives, the current status and the future goals for this project.

Keywords: Laplace Transform, Fourier Transform, Interactive Theorem Proving, HOL Light

1 Introduction

Differential equations are indispensable for modeling the dynamical behaviour of continuous-time engineering and physical systems. Transform methods, which include the Laplace and Fourier transform, have been widely used for the differential equation based dynamical analysis of these systems. These transform methods are the integral based methods, which convert a time varying function into its corresponding s or ω-domain representations based on Laplace and Fourier transform, respectively. Moreover, this transformation converts the differential and integral operators in the time domain to their corresponding algebraic operators, namely, multiplication and division, in Laplace (s) or Frequency (ω) domain and thus the arithmetic manipulation of the resulting equations involving these operators becomes easier. These equivalent representations of the differential equations can further be used for the transfer function and frequency response analysis of these continuous-time systems. Laplace transform is used for the analysis of the systems with causal input, whereas, in the case of non-causal input systems, Fourier transform is used. This analysis varies in its complexity depending on the size, design parameters, constraints and the nature of the input and output signals. The Laplace and Fourier transform methods have been widely used for the analysis of many continuous-time systems, as shown in Table 1.

Table 1: Applications of Transform Methods

Laplace Transform	Fourier Transform
	Analog Circuits [6, 15]
Control Systems [1, 2, 3, 4, 5]	Signal Processing [16, 17, 18, 19]
Analog Circuits [6]	Image Processing [20]
Power Electronics [7, 8]	Biomedical Imaging [18]
Astronomy [9, 10, 11]	Communication systems [21, 22, 23, 24]
Mechanical Systems [12, 5]	Mechanical Systems [12]
Nuclear Physics [13, 14]	Optics [25, 26]
	Electromegnatics [27, 28, 29]

Traditionally, the transform methods based analysis is done using paper-and-pencil proof and computer simulation methods, such as symbolic and numerical methods. However, due to the human-error proneness of paper-and-pencil proof methods and the presence of unverified symbolic algorithms, discretization errors and numerical errors in the simulations methods, the accuracy of the analysis cannot be ascertained. This in turn can lead to compromising performance and efficiency of the underlying system. Interactive theorem proving [30] allows us to overcome these limitations by providing support for logic-based modeling of the system and its intended behaviour and verifying their relationship based on deductive reasoning within the sound core of a theorem prover. With the same motivation, the Laplace [31] and Fourier [32] transforms have been formalized in

higher-order logic. In the report, we mainly describe the past, ongoing and the planned activities for this project[1], which was started in System Analysis and Verification (SAVe) lab[2] in 2012. The formalization of Laplace [31] and Fourier [32] transforms has been developed using the multivariate calculus theories of HOL Light [33]. These formalizations also include the formal verification of some of the classical properties, such as, existence, linearity, frequency shifting, modulation, time reversal, differentiation and integration in time-domain. We choose HOL Light theorem prover for the transform methods based analysis due to the presence of the multivariate calculus theories [33], which contain an extensive reasoning support for differential, integral, transcendental and topology theories.

The rest of the report is organised as follows: Section 2 presents the related work. The proposed approach for the transform methods based analysis is presented in Section 3. We present the mathematical and formal definitions of transform methods, some of their formally verified classical properties and their mutual relationship in Section 4. Section 5 provides the detail about the tasks that have been completed so far, the challenges faced during the formalization of transform methods, the current status and the future goals in this project. We present some of the case studies to illustrate the usefulness of the formal transform based analysis in Section 6. Finally, Section 7 concludes the report.

2 Related Work

Fast Fourier transform (FFT) is used for the computation of discrete Fourier transform (DFT), which is used for the analysis of the systems with the discrete-time input. Theorem provers, such as ACL2, HOL and PVS have been used for the verification of different FFT algorithms. Gamboa [34, 35] mechanically verified the correctness of FFT using a simple proof of FFT proposed by Misra using the ACL2 theorem prover. This proof utilizes the powerlist data structures, which enable the modeling of FFT using recursive functions in an efficient way and can handle the verification of many complex FFT algorithms. Similarly, Capretta [36] formalized the FFT and inverse Fourier transform (iFT) in Coq. The author used structural recursion to formalize the FFT. Whereas, the iFT is formalized using a different data type to facilitate formal reasoning about the summation operation. Moreover, isomorphism is used to link both of these data types. Similarly, Akbarpour et al. [37] used the HOL theorem prover for the formal specification and verification of a generic FFT algorithm. The authors used real, complex, IEEE floating point and fixed-point arithmetic theories of HOL to perform the error analysis of the FFT algorithms at real, floating-point and fixed-point levels.

Harrison [38] formalized the Fourier series for a real-valued function in the HOL Light theorem prover. The formalization includes the formal definition of Fourier series and formal verification of some of its properties. Similarly, Chau et al. [39] formalized the Fourier coefficient formulas and their properties in ACL2(r). Fourier series and their formalizations presented in HOL Light and ACL2(r) can only cater for the systems with inputs represented as periodic functions. Recently, Z-transform [40] has also been formalized in the HOL Light theorem prover. However, Z-transform can only be utilized for the analysis of the systems with discrete-time input functions and cannot cater for the continuous-time systems, which is the main focus of the current report.

3 Proposed Approach

Fig. 1 depicts the proposed approach for the transform methods based analysis of the continuous-time systems using the HOL Light theorem prover. The user provides the differential equation that models the dynamics of the system, which needs to be analyzed, and the corresponding input to the system. This differential equation is modeled in higher-order logic using the multivariate calculus theories of HOL Light. In the next step, we need to verify the corresponding properties, such as transfer function, frequency response or the solution of the corresponding differential equations. Our formalization of the Laplace and Fourier transform methods is used to develop the formal reasoning related to this verification. Our formal approach allows the user to perform the analysis of a continuous-time system by selecting the suitable transform method (Laplace or Fourier) depending on the type of the system's input, i.e., if the input to the system is a causal function, then the Laplace transform is used. Similarly, in the case of the non-causal input, Fourier transform can be used.

[1] http://save.seecs.nust.edu.pk/projects/tm.html

[2] http://save.seecs.nust.edu.pk

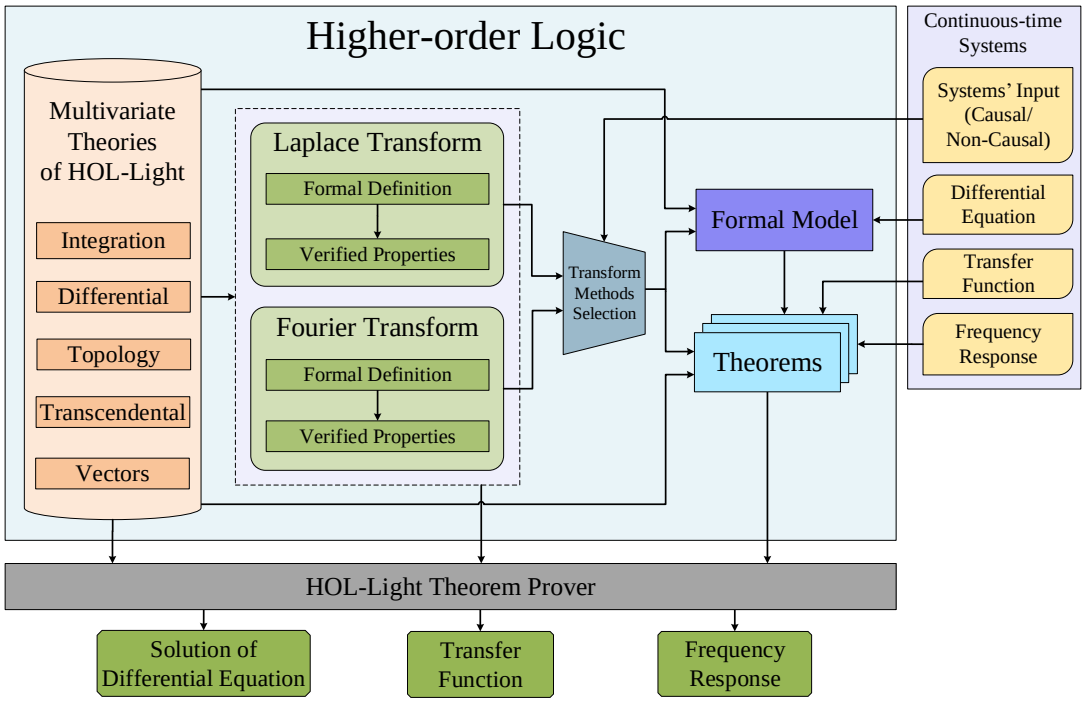


Figure 1: Transform Methods based Formal Analysis

4 Results and Discussions

In this section, we present the existing formal definitions and some of the formally verified classical properties of the Laplace and Fourier transforms. We also give some suggestions that can improve these definitions in terms of reasoning effort required to verify their properties.

4.1 Laplace Transform

Laplace transform of a function $f(t) : \mathbb{R}^1 \to \mathbb{C}$ is mathematically expressed as the following equation [9]:

$$\mathcal{L}[f(t)] = (\mathcal{L}f)(s) = F(s) = \int_0^\infty f(t)e^{-st}dt,\ s \in \mathbb{C} \tag{1}$$

where s is a complex variable. The limit of integration is from 0 to ∞. The above equation can alternatively represented as:

$$F(s) = \lim_{b\to\infty} \int_0^b f(t)e^{-st}dt \tag{2}$$

We formalize Equation 1 using its alternate representation (Equation 2), as follows [31]:

Definition 4.1. Laplace Transform

```
⊢ ∀ s f.  laplace f s =
    lim at_posinfinity (λb.  integral (interval [lift (&0), lift b])
                             (λt.  cexp (--(s * Cx (drop t))) * f t))
```

In the above definition, `integral` represents the vector integral. It takes the integrand function `f` : $\mathbb{R}^N \to \mathbb{R}^M$, and a vector-space `i` : $\mathbb{R}^N \to \mathbb{B}$, which defines the region of convergence, and returns the integral of `f` on `i` as a vector $\mathbb{R}^M$ [41]. The function `lim` in Definition 4.1 takes a vector function `f` : $\mathbb{A} \to \mathbb{R}^M$ and `net` : $\mathbb{A}$ and returns `l` of data-type $\mathbb{R}^M$, i.e., the value to which `f` converges at the given `net`. The function `lift` accepts a variable of type $\mathbb{R}$ and maps it to a 1-dimensional vector with the input variable as its single component. Similarly, `drop` takes a 1-dimensional vector and returns its single element as a real number [42].

The Laplace transform of a function f exists, if the function `f` is piecewise smooth and of exponential order on the positive real line [9]. The existence of the Laplace transform is formally defined as follows [31]:

Definition 4.2. Laplace Exists

```
⊢ ∀ s f.  laplace_exists f s ⇔
     (∀ b.  f piecewise_differentiable_on interval [lift (&0), lift b]) ∧
     (∃ M a.  Re s > drop a ∧ exp_order f M a)
```

The function `exp_order` in the above definition is formally defined as [31]:

Definition 4.3. Exponential Order Function

```
⊢ ∀ f M a.  exp_order f M a ⇔ &0 < M ∧
              (∀ t.  &0 <= t ⇒ norm (f (lift t)) <= M * exp (drop a * t))
```

We used Definitions 4.1, 4.2 and 4.3 to formally verify some of the classical properties of the Laplace transform, given in Table 2, which mainly include the linearity, frequency shifting, differentiation and integration in the time domain. The formalization of the Laplace transform took around 5000 lines of code and approximately 450 man-hours.

Table 2: Properties of Laplace Transform

Mathematical Form	Formalized Form
Limit Existence of Integral of Laplace Transform	
$\exists l.\ \left(\int_0^\infty f(t)e^{-st} \to l\right)$	`⊢ ∀ f s.  laplace_exists f s ⇒ (∃l.  ((λb.  integral (interval [lift (&0),lift b]) (λt.  cexp (--(s * Cx (drop t))) * f t)) → l) at_posinfinity)`
Linearity	
$\mathcal{L}[\alpha f(t) + \beta g(t)] = \alpha F(s) + \beta G(s)$	`⊢ ∀ f g s a b.  laplace_exists f s ∧ laplace_exists g s ⇒ laplace (λt.  a * f t + b * g t) s = a * laplace f s + b * laplace g s`
Frequency Shifting	
$\mathcal{L}[e^{s_0 t} f(t)] = F(s - s_0)$	`⊢ ∀ f s s0.  laplace_exists f s ⇒ laplace (λt.  cexp (s0 * Cx (drop t)) * f t) s = laplace f (s - s0)`
First-order Differentiation	
$\mathcal{L}\left[\frac{d}{dt}f(t)\right] = sF(s) - f(0)$	`⊢ ∀ f s.  laplace_exists f s ∧ (∀t.  f differentiable at t) ∧ laplace_exists (λt.  vector_derivative f (at t)) s ⇒ laplace (λt.  vector_derivative f (at t)) s = s * laplace f s - f (lift (&0))`
Higher-order Differentiation	
$\mathcal{L}[\frac{d^n}{dt^n}f(t)] = s^n F(s) - \sum_{k=1}^{n} s^{k-1}\frac{d^{n-k}f(0)}{dx^{n-k}}$	`⊢ ∀ f s n.  laplace_exists_higher_deriv n f s ∧ (∀t.  higher_derivative_differentiable n f t) ⇒ laplace (λt.  higher_vector_derivative n f t) s = s pow n * laplace f s - vsum (1..n) (λk.  s pow (k - 1) * higher_vector_derivative (n - k) f (lift (&0)))`
Integration in Time Domain	
$\mathcal{L}\left[\int_0^t f(\tau)d\tau\right] = \frac{1}{s}F(s)$	`⊢ ∀ f s.  &0 < Re s ∧ laplace_exists f s ∧ laplace_exists (λx.  integral (interval [lift (&0),x]) f) s ∧ (∀x.  f continuous_on interval [lift (&0),x]) ⇒ laplace (λx.  integral (interval [lift (&0),x]) f) s = inv s * laplace f s`

The formal definition of the Laplace transform presented as Definition 4.1 is modeled using the notion of the limit. However, the HOL Light definition of the integral function (`integral`) implicitly encompasses infinite limits of integration, so we do not require to include another limiting process in its definition. Moreover, the region of integration (the positive real line) given in Equation 1 can be modeled using the notion of set. So the mathematical definition of the Laplace transform, given by Equation 1, can alternatively be modeled in HOL Light as:

```
⊢ ∀ s f.  laplace_transform f s = integral {t| &0 <= drop t} (λt.  cexp (--(s * Cx (drop t))) * f t)
```

In the above definition, the region of integration, i.e., $[0, \infty)$ is modeled as `{t | &0 <= drop t}` and this definition is equivalent to Definition 4.1. Moreover, this revised definition considerably simplifies the reasoning

process in the verification of the properties of the Laplace transform since it does not involve the notion of limit.

4.2 Fourier Transform

The Fourier transform of a function $f(t) : \mathbb{R}^1 \to \mathbb{C}$ is mathematically defined as:

$$\mathcal{F}[f(t)] = (\mathcal{F}f)(\omega) = F(\omega) = \int_{-\infty}^{+\infty} f(t)e^{-i\omega t}dt,\ \omega \in \mathbb{R} \tag{3}$$

where ω is a real variable. The limit of integration is from $-\infty$ to $+\infty$. We formalize Equation (3) as the following HOL Light function [32]:

Definition 4.4. Fourier Transform

```
⊢ ∀ w f.  fourier f w = integral UNIV (λt.  cexp (--((ii * Cx w) * Cx (drop t))) * f t)
```

The Fourier transform of a function f exists, i.e., the integrand of Equation 3 is integrable, and the integral has some converging limit value, if f is piecewise smooth and is absolutely integrable on the whole real line [9, 32]. The Fourier existence condition can thus be formalized in HOL Light as follows:

Definition 4.5. Fourier Exists

```
⊢ ∀ f.  fourier_exists f =
   (∀ a b.  f piecewise_differentiable_on interval [lift a, lift b]) ∧
            f absolutely_integrable_on {x | &0 <= drop x} ∧
            f absolutely_integrable_on {x | drop x <= &0}
```

In the above function, the first conjunct expresses the piecewise smoothness condition for the function `f`. Whereas, the next two conjuncts represent the condition that the function `f` is absolutely integrable on the whole real line.

We used Definitions 4.4 and 4.5 to verify some of the classical properties of Fourier transform, given in Table 3, such as existence, linearity, frequency shifting, modulation, time reversal and differentiation in time-domain. The formalization took around 3000 lines of code and approximately 250 man-hours.

Table 3: Properties of Fourier Transform

Mathematical Form	Formalized Form
Integrability	
$f(t)e^{-i\omega t}$ *integrable on* $(-\infty, \infty)$	`⊢ ∀ f w.  fourier_exists f s ⇒ (λt.  cexp (--((ii * Cx w) * Cx (drop t))) * f t) integrable_on UNIV`
Linearity	
$\mathcal{F}[\alpha f(t) + \beta g(t)] = \alpha F(\omega) + \beta G(\omega)$	`⊢ ∀ f g w a b.  fourier_exists f ∧ fourier_exists g ⇒ fourier (λt.  a * f t + b * g t) w = a * fourier f w + b * fourier g w`
Frequency Shifting	
$\mathcal{F}[e^{i\omega_0 t}f(t)] = F(\omega - \omega_0)$	`⊢ ∀ f w w0.  fourier_exists f ⇒ fourier (λt.  cexp ((ii * Cx w0) * Cx (drop t)) * f t) w = fourier f (w - w0)`
Modulation (Cosine and Sine Based Modulation)	
$\mathcal{F}[cos(\omega_0 t)f(t)] = \frac{F(\omega-\omega_0)+F(\omega+\omega_0)}{2}$	`⊢ ∀ f w w0.  fourier_exists f ⇒ fourier (λt.  ccos (Cx w0 * Cx (drop t)) * f t) w = (fourier f (w − w0) + fourier f (w + w0)) / Cx (&2)`
$\mathcal{F}[sin(\omega_0 t)f(t)] = \frac{F(\omega-\omega_0)-F(\omega+\omega_0)}{2i}$	`⊢ ∀ f w w0.  fourier_exists f ⇒ fourier (λt.  csin (Cx w0 * Cx (drop t)) * f t) w = (fourier f (w − w0) − fourier f (w + w0)) / Cx (&2) * ii`
Time Reversal	
$\mathcal{F}[f(-t)] = F(-\omega)$	`⊢ ∀ f w.  fourier_exists f ⇒ fourier (λt.  f (--t)) w = fourier f (--w)`
First-order Differentiation	

$\mathcal{F}[\frac{d}{dt}f(t)] = i\omega F(\omega)$	⊢ ∀ f w. fourier_exists f ∧ fourier_exists (λt. vector_derivative f (at t)) ∧ (∀t. f differentiable at t) ∧ ((λt. f (lift t)) → vec 0) at_posinfinity ∧ ((λt. f (lift t)) → vec 0) at_neginfinity ⇒ fourier (λt. vector_derivative f (at t)) w = ii ∗ Cx w ∗ fourier f w
Higher-order Differentiation	
$\mathcal{F}[\frac{d^n}{dt^n}f(t)] = (i\omega)^n F(\omega)$	⊢ ∀ f w n. fourier_exists_higher_deriv n f ∧ (∀t. differentiable_higher_derivative n f t) ∧ (∀p. p < n ⇒ ((λt. higher_vector_derivative p f (lift t)) → vec 0) at_posinfinity) ∧ (∀p. p < n ⇒ ((λt. higher_vector_derivative p f (lift t)) → vec 0) at_neginfinity) ⇒ fourier (λt. higher_vector_derivative n f t) w = (ii ∗ Cx w) pow n ∗ fourier f w

The absolute integrability condition in Definition 4.5 is modeled using two conjuncts, i.e., the absolute integrability on the positive and negative real line, respectively. This condition can alternatively be modeled as:

```
f absolutely_integrable_on UNIV
```

The function `UNIV` in the above condition presents the whole real line and is a composite modeling of the positive and negative real lines. Thus, this revised condition can better model the integrability condition and its equivalence to the earlier condition can be easily verified using some properties of the integrals.

4.3 Relationship between Laplace and Fourier Transforms

By restricting the complex-valued function $f(t) : \mathbb{R}^1 \to \mathbb{C}$ and the Laplace variable $s : \mathbb{R}^2$, we can find a very important relationship between Laplace and Fourier transforms. If the function f is causal, i.e., $f(t) = 0$ for all $t < 0$ and the real part of the Laplace variable $\mathtt{s} : \mathtt{R}^2$ is zero, i.e., *Re s = 0*, then the Laplace transform of function f is equal to Fourier transform [6]:

$$(\mathcal{L}f)(s)\,|_{Re\ s\ =\ 0} = \ (\mathcal{F}f)(Im\ s)$$

The above relationship can be verified in HOL Light as follow:

```
⊢ ∀ f s. laplace_exists f s ∧ (∀t. t IN {t | drop t <= &0} ⇒ f t = vec 0) ∧ (∀t. Re s = &0)
                  ⇒ laplace_transform f s = fourier_transform f (Im s)
```

This relationship is very crucial in a sense, if the function is causal, then the Laplace transform can be used in the analysis, rather than the Fourier transform.

5 Achieved Goals, Current Status and Future Plans

The project started with the formalization of the Laplace transform and one of the major challenge faced during its formalization was that we were not very familiar with multivariable calculus theories of HOL Light and thus reasoning about theorems involving integration, differentiation and limits of the real and vector functions was very tedious for us as novice users of the system. Moreover, we found that many basic properties required to reason about transform methods were not available in the multivariable theories in HOL Light and thus we ended up verifying many classical properties related to integration, differentiation and limit, including *Comparison test for improper integrals*, *Integration by substitution*, *Integration by parts* and the *Relationship between derivative of a real and vector functions* [31]. The other major difficulty faced during these formalizations was the unavailability of detailed proofs for the properties of transform methods in literature. The available paper-and-pencil based proofs were found to be very abstract and we had to build the formal reasoning, at our own, for their verification. Moreover, some of the assumptions of the properties of the Fourier transform were not explicitly mentioned in the literature, which we have extracted during the verification of these properties. The foundational formalization

of the Laplace and Fourier transform took about 8000 lines of HOL Light code and 700 man hours. The main benefit of this formalization was found in the ability to conduct formal transform method based analysis of many systems, including linear transfer converter [31], which is widely used component in power electronics, the first and second-order Sallen-Key low-pass filters [43] and the automobile suspension system [32]. The foundational formalization of Laplace and Fourier transforms was found to be quite useful in this context and the analysis of these applications was found to be very straightforward and took only 1600 lines of HOL Light code and 8 man hours only.

The distinguishing feature of the transform methods based formal analysis, compared to traditional analysis methods, is the generic nature of the formally verified theorems. All the variables and functions are universally quantified and thus can be specialized to obtain the results for any given values. Moreover, all of the required assumptions are guaranteed to be explicitly mentioned along with a formally verified theorem due to the inherent soundness of the theorem proving approach. Moreover, the high expressiveness of the higher-order logic enables us to model the differential equation and the corresponding transfer function and frequency response in their true continuous form, whereas, in model checking, they are mostly discretized and modeled using a state-transition system, which may compromise the completeness of the analysis. A comparison of different analysis techniques for the transform methods is presented in Table 4. The evaluation of these techniques is performed based on various parameters, such as expressiveness, accuracy and automation. For example, in model checking, we cannot truly model the integration and differentiation, and their discretization results into an abstracted model, which makes it less expressive. Moreover, in theorem proving, the verification is done interactively due to the undecidable nature of higher-order logic. We are mainly working on facilitating the user in this interactive verification part by providing formal reasoning support for Laplace and Fourier transforms.

Table 4: Comparison of Analysis Techniques for Transform Methods

	Paper-and-Pencil Proof	Simulation	Computer Algebra System	Model Checking	Theorem Proving
Expressiveness	✓	✓	✓		✓
Accuracy	✓(?)			✓	✓
Automation		✓	✓	✓	

We are currently focussing on the following three tasks:

- Formalization of inverse Laplace and Fourier transforms: We have formally verified the uniqueness property of the Laplace transform and are working on verifying it for the Fourier Transform. These properties would enable us to verify the analytical solutions of linear differential equations.
- Automation of the transform methods based formal analysis: We are in the process of developing some tactics to automate the transform methods based formal analysis of the continuous-time systems. These tactics would only require the differential equation, modeling the dynamics of the systems, and expressions for the corresponding transfer functions and frequency responses and would automatically verify the relationships between them. This would allow non-experts in theorem proving to benefit from our formal approach for the analysis of the systems.
- Formalization of Vectorial Laplace transform: The current formalization of the Laplace transform can only be used for the formal analysis of the single-input single-output (SISO) control systems. We are working on extending the reasoning support for the normal Laplace transform to complex vectors. The resulting formalization would help us to formally verify the transfer function of the multiple-input multiple output (MIMO) control systems, which are modeled using the state space representations.

To further extend the scope of transform methods based formal analysis of systems, we plan to work on the following two tasks in the future:

- Linking the formal library of the Laplace transform with the formalization of Z-transform [40]: This linkage will enable us to perform the formal analysis of the hybrid (exhibiting continuous and discrete behaviour) systems.
- Formalization of two-dimensional Fourier transform: This requires the formalization of double integral and its properties, which, to the best of our knowledge, have not been formalized in the current multivariable calculus theories of HOL Light. The two-dimensional Fourier transform would build upon this theory of double integration. This formalization will enable us to perform the formal analysis of many electromagnetic

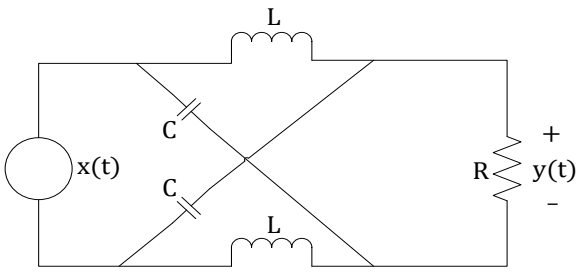
(e.g., [44, 45]) and the optical systems (e.g., [46, 44]). Moreover, this formalization of double integral can also be used for the formal analysis of some other applications in physics, such as, quantum [47] and mechanics [48].

6 Impact

Our foundational formalization of the Laplace [31] and the Fourier [32] transforms has been used for the formal analysis of the various systems and some of them are presented in Table 5. Due to the availability of the higher-order-logic

Table 5: Applications of Transform Methods

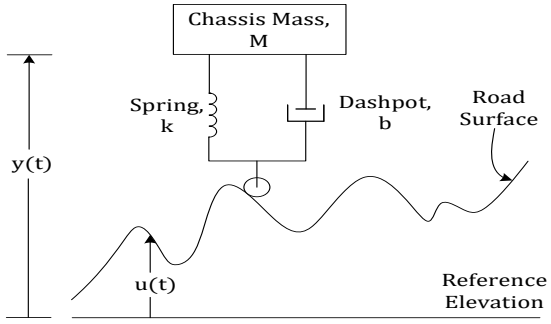
Linear Transfer Converter [31]



Lines of code: 650
Man-hours: 2

$$\vdash \forall\ \texttt{y u s R L C.}\ (\texttt{\&0 < R}) \wedge (\texttt{\&0 < L}) \wedge (\texttt{\&0 < C}) \wedge \texttt{zero\_initial\_conditions 1 u} \wedge \texttt{zero\_initial\_conditions 1 y} \wedge \texttt{(higher\_derivative\_laplace\_exists 2 u s)} \wedge \texttt{(higher\_derivative\_laplace\_exists 2 y s)} \wedge (\forall \texttt{t.}\ \texttt{higher\_derivative\_differentiable 2 u t}) \wedge (\forall \texttt{t.}\ \texttt{higher\_derivative\_differentiable 2 y t}) \wedge (\forall \texttt{t.}\ \texttt{diff\_eq\_LTC y u L C R}) \wedge \texttt{(non\_zero\_denominator u s R L C)} \Rightarrow \frac{\texttt{laplace y s}}{\texttt{laplace x s}} = \frac{\texttt{s pow 2} - \texttt{Cx}\left(\frac{\texttt{\&1}}{\texttt{L} * \texttt{C}}\right)}{\texttt{Cx}\left(\frac{\texttt{\&1}}{\texttt{L} * \texttt{C}}\right) - \texttt{Cx}\left(\frac{\texttt{\&2}}{\texttt{R} * \texttt{C}}\right) * \texttt{s} + \texttt{s pow 2}}$$

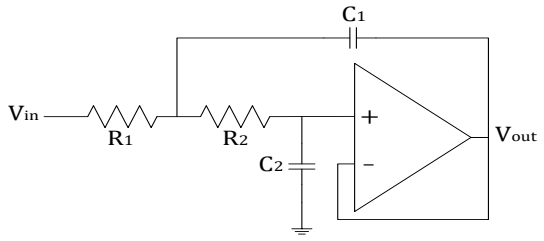
Automobile Suspension System [32]



Lines of code: 500
Man-hours: 2

$$\vdash \forall\ \texttt{y u w a.}\ (\texttt{\&0 < M}) \wedge (\texttt{\&0 < b}) \wedge (\texttt{\&0 < k}) \wedge (\forall \texttt{t.}\ \texttt{differentiable\_higher\_derivative 2 y t}) \wedge (\forall \texttt{t.}\ \texttt{differentiable\_higher\_derivative 1 u t}) \wedge \texttt{(fourier\_exists\_higher\_deriv 2 y)} \wedge \texttt{(fourier\_exists\_higher\_deriv 1 u)} \wedge (\forall \texttt{p.}\ \texttt{p < 2} \Rightarrow ((\lambda \texttt{t.}\ \texttt{higher\_vector\_derivative p y (lift t)}) \rightarrow \texttt{vec 0}) \texttt{ at\_posinfinity})) \wedge (\forall \texttt{p.}\ \texttt{p < 2} \Rightarrow ((\lambda \texttt{t.}\ \texttt{higher\_vector\_derivative p y (lift t)}) \rightarrow \texttt{vec 0}) \texttt{ at\_neginfinity})) \wedge ((\lambda \texttt{t.}\ \texttt{u (lift t)}) \rightarrow \texttt{vec 0}) \texttt{ at\_posinfinity}) \wedge ((\lambda \texttt{t.}\ \texttt{u (lift t)}) \rightarrow \texttt{vec 0}) \texttt{ at\_neginfinity}) \wedge (\forall \texttt{t.}\ \texttt{diff\_eq\_ASS y u b M k}) \wedge \texttt{(non\_zero\_denominator u w b M k)} \Rightarrow \frac{\texttt{fourier y w}}{\texttt{fourier x w}} = \frac{\texttt{Cx}\left(\frac{\texttt{b}}{\texttt{M}}\right) * \texttt{ii} * \texttt{Cx w} + \texttt{Cx}\left(\frac{\texttt{k}}{\texttt{M}}\right)}{\texttt{Cx}\left(\frac{\texttt{k}}{\texttt{M}}\right) + \texttt{Cx}\left(\frac{\texttt{b}}{\texttt{M}}\right) * \texttt{ii} * \texttt{Cx w} + (\texttt{ii} * \texttt{Cx w})\ \texttt{pow 2}}$$

Second order Sallen-key Filter [43]



Lines of code: 250
Man-hours: 2

$$\vdash \forall \texttt{R1 R2 C1 C2 Vin Vout s.}\ (\&0 < \texttt{R1}) \wedge (\&0 < \texttt{C1}) \wedge (\&0 < \texttt{C1}) \wedge (\&0 < \texttt{C2}) \wedge \texttt{zero\_initial\_conditions Vin Vout Va} \wedge (\texttt{laplace\_exists\_higher\_deriv 2 Vout s}) \wedge (\texttt{laplace\_exists\_higher\_deriv 2 Vin s}) \wedge (\forall \texttt{t. differentiable\_higher\_derivative 2 Vout t}) \wedge (\forall \texttt{t. differentiable\_higher\_derivative 2 Vin t}) \wedge (\forall \texttt{t. differentiable\_higher\_derivative 2 Va t}) \wedge (\texttt{non\_zero\_denom Vin s R1 R2 C1 C2}) \wedge (\forall \texttt{t. SKF\_behav Vin Vout R1 R2 C1 C2}) \Rightarrow \frac{\texttt{laplace Vout s}}{\texttt{laplace Vin s}} = \frac{\texttt{Cx}(\&1)}{\texttt{Cx}\left(\texttt{R1} * \texttt{C1} * \texttt{R2} * \texttt{C2}\right) * \texttt{s pow 2} + \texttt{Cx}\left(\texttt{C2} * (\texttt{R1} + \texttt{R2})\right) * \texttt{s} + \texttt{Cx}(\&1)}$$

formalization of transform methods, the analysis of these applications was very straightforward. It can be seen that these analyses took very few lines of code and very less manual effort, which clearly illustrates the effectiveness of our foundational formalization. These formalizations of the Laplace and Fourier transform can be further used for the analysis of the many other applications, including control systems, power electronics, signal processing and communication systems.

7 Conclusion

This report provides a synthetic presentation of our ongoing project on the formalization of transform methods using the HOL Light theorem prover. We present the proposed approach for the transform methods based formal analysis of the continuous-time systems along with the foundational formal definitions of the Laplace and Fourier transform. The report highlights the main objectives of the project that have been achieved so far, the challenges faced during this formalization, and the ongoing tasks and the future goals for this project. Once all the planned formalization tasks are accomplished, then these foundations can be used for the formal analysis of many safety-critical systems, such as control systems, power electronics, signal processing, electromagnetics and optical systems.

References

[1] Katsuhiko Ogata and Yanjuan Yang. Modern Control Engineering. 1970.

[2] Norman S Nise. *Control Systems Engineering.* John Wiley & Sons, 2007.

[3] Richard C Dorf and Robert H Bishop. Modern Control Systems. 1998.

[4] Thomas E Fortmann and Konrad L Hitz. *An introduction to linear control systems.* Crc Press, 1977.

[5] Theodore F Bogart. *Laplace Transforms and Control Systems Theory for Technology: Including Microprocessor-based Control Systems.* John Wiley & Sons, 1982.

[6] Roland E Thomas, Albert J Rosa, and Gregory J Toussaint. *The Analysis and Design of Linear Circuits, Binder Ready Version.* John Wiley & Sons, 2016.

[7] Muhammad H Rashid. *Power Electronics: Circuits, Devices, and Applications.* Pearson Education India, 2009.

[8] Gonzalo Abad. *Power Electronics and Electric Drives for Traction Applications.* John Wiley & Sons, 2016.

[9] R. J. Beerends, H. G. Morsche, J. C. Van den Berg, and E. M. Van de Vrie. *Fourier and Laplace Transforms.* Cambridge University Press, 2003.

[10] William E Boyce, Richard C DiPrima, and Charles W Haines. *Elementary Differential Equations and Boundary Value Problems*, volume 9. Wiley New York, 1969.

[11] Joseph M Hilbe. *Astrostatistical Challenges for the New Astronomy*, volume 1. Springer Science & Business Media, 2012.

[12] Alan V Oppenheim, Alan S Willsky, and S Hamid Nawab. *Signals and Systems*. Prentice Hall Processing series. Prentice Hall, Inc., 2 edition, 1996.

[13] Norman William McLachlan. *Laplace Transforms and their Applications to Differential Equations*. Courier Corporation, 2014.

[14] Weston M Stacey. *Nuclear Reactor Physics*. John Wiley & Sons, 2007.

[15] William McC Siebert. *Circuits, Signals, and Systems*, volume 2. MIT press, 1986.

[16] Athanasios Papoulis. *Signal Analysis*, volume 2. McGraw-Hill, 1977.

[17] Patrick Gaydecki. *Foundations of Digital Signal Processing: Theory, Algorithms and Hardware Design*. Institution of Engineering and Technology, 2004.

[18] Suresh R Devasahayam. *Signals and Systems in Biomedical Engineering: Signal Processing and Physiological Systems Modeling*. Springer Science & Business Media, 2012.

[19] Eleanor Chu. *Discrete and Continuous Fourier Transforms: Analysis, Applications and Fast Algorithms*. Crc Press, 2008.

[20] Geoff Dougherty. *Digital Image Processing for Medical Applications*. Cambridge University Press, 2009.

[21] Rodger Ziemer and William H Tranter. *Principles of Communications: System Modulation and Noise*. John Wiley & Sons, 2006.

[22] Ke-Lin Du and M. N. S Swamy. *Wireless Communication Systems: from RF Subsystems to 4G Enabling Technologies*. Cambridge University Press, 2010.

[23] Upamanyu Madhow. *Introduction to Communication Systems*. Cambridge University Press, 2014.

[24] Lyman Chapin. Communication Systems. 1978.

[25] Jack D Gaskill. Linear Systems, Fourier Transforms, and Optics. *Linear Systems, Fourier Transforms, and Optics by Jack D. Gaskill John Wiley and Sons*, 1, 1978.

[26] Henry Stark. *Application of Optical Fourier Transforms*. Elsevier, 2012.

[27] David B Davidson. *Computational Electromagnetics for RF and Microwave Engineering*. Cambridge University Press, 2005.

[28] Bernard Jancewicz. Trivector Fourier Transformation and Electromagnetic Field. *Journal of mathematical physics*, 31(8):1847–1852, 1990.

[29] Epameinondas E Kriezis, DP Chrissoulidis, and AG Papagiannakis. *Electromagnetics and Optics*. World Scientific, 1992.

[30] Osman Hasan and Sofiène Tahar. Formal Verification Methods. *Encyclopedia of Information Science and Technology, IGI Global Pub*, pages 7162–7170, 2015.

[31] S. H. Taqdees and O. Hasan. Formalization of Laplace Transform Using the Multivariable Calculus Theory of HOL-Light. In *Logic for Programming, Artificial Intelligence, and Reasoning*, volume 8312 of *LNCS*, pages 744–758. Springer, 2013.

[32] Adnan Rashid and Osman Hasan. On the Formalization of Fourier Transform in Higher-order Logic. In *Interactive Theorem Proving*, volume 9807 of *LNCS*, pages 483–490. Springer, 2016.

[33] John Harrison. HOL Light Multivariate Calculus. `https://github.com/jrh13/hol-light/tree/master/Multivariate`, 2017.

[34] Ruben Gamboa. Mechanically Verifying the Correctness of the Fast Fourier Transform in ACL2. *Parallel and Distributed Processing*, pages 796–806, 1998.

[35] Ruben A Gamboa. The Correctness of the Fast Fourier Transform: A Structured Proof in ACL2. *Formal Methods in System Design*, 20(1):91–106, 2002.

[36] Venanzio Capretta. Certifying the Fast Fourier Transform with Coq. In *International Conference on Theorem Proving in Higher Order Logics*, pages 154–168. Springer, 2001.

[37] Behzad Akbarpour and Sofiene Tahar. A Methodology for the Formal Verification of FFT Algorithms in HOL. In *International Conference on Formal Methods in Computer-Aided Design*, pages 37–51. Springer, 2004.

[38] John Harrison. Fourier Series. `http://github.com/jrh13/hol-light/blob/master/100/fourier.ml`, 2015.

[39] Cuong K Chau, Matt Kaufmann, and Warren A Hunt Jr. Fourier Series Formalization in ACL2 (r). *arXiv preprint arXiv:1509.06087*, 2015.

[40] Umair Siddique, Mohamed Yousri Mahmoud, and Sofiene Tahar. On the Formalization of Z-Transform in HOL. In *Interactive Theorem Proving*, volume 8558 of *LNCS*, pages 483–498. Springer, 2014.

[41] John Harrison. Integration Theory in HOL Light. `https://github.com/jrh13/hol-light/blob/master/Multivariate/integration.ml`, 2017.

[42] John Harrison. Real Vectors in Euclidean Space. `http://github.com/jrh13/hol-light/blob/master/Multivariate/vectors.ml`, 2017.

[43] S. H. Taqdees and O. Hasan. Formally Verifying Transfer Functions of Linear Analog Circuits. IEEE Design & Test, `http://save.seecs.nust.edu.pk/pubs/2017/DTnA_2017.pdf`, 2017.

[44] R. N. Bracewell. *The Fourier Transform and its Applications*. McGraw-Hill, 1978.

[45] Jian-Ming Jin. *Theory and Computation of Electromagnetic Fields*. John Wiley & Sons, 2011.

[46] Max Born and Emil Wolf. *Principles of Optics: Electromagnetic Theory of Propagation, Interference and Diffraction of Light*. Elsevier, 1980.

[47] Vittorio Gorini and Alberto Frigerio. *Fundamental aspects of quantum theory*, volume 144. Springer Science & Business Media, 2012.

[48] Andrew Pytel and Jaan Kiusalaas. *Engineering Mechanics: Dynamics*. Nelson Education, 2016.