Journal of Scientific Computing and Machine Learning, vol. 1, pp. 60–67, 2025
Submission Date: January 16, 2025 Acceptance Date: January 26, 2025

An Infinite Dimensional LSSL with Infinite Dimensional HiPPO

Atsushi Takabatake[1] and Takaharu Yaguchi[2]

[1,2]Department of Mathematics, Kobe University, 1-1 Rokkodai-cho, Nada-ku
244s008s@stu.kobe-u.ac.jp

Abstract

A state-space-model-based model using HiPPO matrices has been proposed as an efficient time series modeling method. In this paper, we discuss the operator that is a generalization of the HiPPO matrix to infinite dimension. We also propose an infinite-dimensional version of linear state-space layer (LSSL) using the operator, and in particular, a matrix with an explicit DPLR representation that is derived from it. Then, we describe the results of numerical experiments with the S4 model using the matrix.

Keywords S4Model · DPLR · LSSL · hyperbolic PDE

1. Introduction

Modeling of time-series data requires dealing with strong correlations and long-range dependencies between data that are far from each other in time-series. A machine learning model based on the State Space Model has been proposed as an efficient model that can handle this. In these models, a matrix called HiPPO matrix plays an important role in handling long-range dependencies. Especially, the S4 model is a fast computation model by restricting the form of the parameters required by the Normal Plus Low Rank (NPLR) property of the HiPPO matrix. The S4 model uses the Diagonal Plus Low Rank (DPLR) representation of the NPLR matrix to represent the state-space model. In this case, the DPLR representation is obtained only numerically, and is difficult to obtain analytically.

In this paper, we give an interpretation of the infinite-dimensional version of the linear state-space layer (LSSL) by the HiPPO matrix. In particular, we propose an operator corresponding to the HiPPO matrix, and show that this operator has a DPLR representation. We also present numerical experiments using the matrices derived from it.

2. Background

2.1. LSSL

The model represented by the following ordinary differential equation is called the State Space Model.

$$\frac{dh}{dt}(t) = Ah(t) + Bu(t) \tag{1a}$$

$$y(t) = Ch(t) + Du(t) \tag{1b}$$

where $u : [0, T] \to \mathbb{C}$, $h : [0, T] \to \mathbb{C}^n$, $A \in \mathbb{C}^{n\times n}$, $B \in \mathbb{C}^{n\times 1}$, $C \in \mathbb{C}^{1\times n}$, $D \in \mathbb{C}$. LSSL (Linear State Space Layer) is a discretization of this as a layer in neural networks by seq2seq[2]. In this paper, we consider infinite dimensional LSSL on the function space $L^2([0, N], \mathbb{C})$ instead of $\mathbb{C}^n$ as the latent space.

2.2. S4model

The S4 model is a fast LSSL computation model by restricting the form of A to the NPLR representation defined as follows.

Definition 1 ([1]). *When matrix A is represented by*

$$A = U\Lambda U^* + PQ^* \tag{2}$$

with a unitary matrix $U \in \mathbb{C}^{N\times N}$, a diagonal matrix $\Lambda \in \mathbb{C}^{N\times N}$, and low rank matrices $P, Q \in \mathbb{C}^{N\times M}$ $(M < N)$, this representation is called the NPLR (Normal Plus Low Rank) representation of A. Especially when U is the identity matrix, this representaion is called the DPLR (Diagonal Plus Low Rank) representation of A.

When A has an NPLR representation, the equation of the latent space can be represented by

$$\frac{d(U^*h)}{dt}(t) = (\Lambda + U^*PQ^*U)U^*h(t) + U^*Bu(t), \tag{3}$$

which is a State Space Model using a matrix with a DPLR representation. In the S4 model, A is converted to a matrix with DPLR representation before the calculation.

2.3. HiPPO Matrix

In LSSL, it is known that the accuracy is improved by employing the following HiPPO matrix as the initial value of A:

$$A^{\text{hippo}}_{ij} = -\begin{cases} (2i+1)^{1/2}(2j+1)^{1/2} & (i > j) \\ i+1 & (i = j) \\ 0 & (i < j) \end{cases} \tag{4a}$$

$$B^{\text{hippo}}_i = \sqrt{2i+1}. \tag{4b}$$

In particular, this HiPPO MatrixA^{hippo} has an NPLR representation [1]. However, the proof is not constructive and hence the representation is difficult to obtain directly. In the implementation of [1], the NPLR representation is obtained by numerical calculation.

2.4. Hyperbolic PDE from HiPPO Matrix

LSSL with the HiPPO matrix can be interpreted as a semi-discretization of partial differential equations as follows.

Theorem 1 ([3], [4]). *Let matrices A and B be A^{hippo} and B^{hippo} in LSSL; i.e., consider*

$$\frac{\mathrm{d}\boldsymbol{h}}{\mathrm{d}t} = A^{\text{hippo}}\boldsymbol{h} + B^{\text{hippo}}u. \tag{5}$$

Then this differential equation for $\boldsymbol{h}$ is obtained by semi-discretizing the following partial differential equation by a finite difference method with the step size $\Delta x = 1$:

$$\frac{\partial^2 g}{\partial t \partial x} = -(2x+1)g, \tag{6a}$$

$$g(x, 0) = 0, \tag{6b}$$

$$g(0, t) = \int_0^t u(s)ds, \tag{6c}$$

where the boundary condition (6c) is discretized as

$$g(0, t) = e^{-\Delta xt}\int_0^t e^{\Delta xs}u(s)ds. \tag{7}$$

From this perspective, an infinite-dimensional version of LSSL with HiPPO matrices is discussed in what follows.

3. Infinite Dimensional HiPPO

3.1. Infinite Dimensional View of HiPPO Layer

Consider solving a partial differential equation

$$\frac{\partial^2}{\partial x \partial t}\left(\frac{h}{\sqrt{2x+1}}\right)(x, t) = -\sqrt{2x+1}h(x, t) \tag{8a}$$

$$\frac{\partial h}{\partial t}(0, t) = u(t) \tag{8b}$$

$$h(x, 0) = 0 \tag{8c}$$

under

$$h \in C^2([0, N] \times [0, T]), u \in C([0, T]). \tag{9}$$

By integrating both sides on $[0, x]$ and multiplying by $\sqrt{2x+1}$, we obtain the integral equation

$$\begin{aligned}\frac{\partial h}{\partial t}(x, t) = &-\sqrt{2x+1}\int_0^x \sqrt{2\xi+1}h(\xi, t)d\xi \\ &+ \sqrt{2x+1}u(t)\end{aligned} \tag{10}$$

for $t \in [0,T]$. If we define $h : [0,T] \to L^2([0,N])$ as $h(t)(x) = h(x,t)$, $\mathcal{F}_0 : L^2([0,N]) \to L^2([0,N])$ as

$$\mathcal{F}_0 f = -\sqrt{2x+1}\int_0^x \sqrt{2\xi+1} f(\xi) d\xi \qquad (11)$$

and $\mathcal{G}_0 : \mathbb{C} \to L^2([0,N])$ as

$$\mathcal{G}_0 a = \sqrt{2x+1} a, \qquad (12)$$

the expression (10) can be written as follows.

$$\frac{\partial h}{\partial t}(t) = \mathcal{F}_0 h(t) + \mathcal{G}_0 u(t) \qquad (13)$$

This can be viewed as a state-space model using $L^2([0,N])$ as the latent space. From here, we will consider the following generalization $\mathcal{F} : L^2([0,N]) \to C([0,N]) \subset L^2([0,N])$ for the operator $\mathcal{F}_0$:

$$[\mathcal{F} f](x) = -\chi(x) \int_0^x f(\xi)\overline{\chi(\xi)} d\xi \qquad (14)$$

where $\chi \in C^1([0,N])$. Hereafter, this $\mathcal{F}$ is called Infinite Dimensional HiPPO.

3.2. Infinite Dimensional HiPPO

We will see that the Infinite Dimensional HiPPO has a "DPLR representation" similar to the HiPPO matrix.

Define $\mathcal{G} : \mathbb{C} \to C([0,N]) \subset L^2([0,N])$ as

$$\mathcal{G} a = a\chi, \qquad (15)$$

$\mathcal{H} : L^2([0,N]) \to \mathbb{C}$ as

$$\begin{aligned}\mathcal{H} f &= \langle f, \chi \rangle_{L^2([0,N])} \\ &= \int_0^N f(\xi)\overline{\chi(\xi)} d\xi\end{aligned} \qquad (16)$$

and $\mathcal{J} : L^2([0,N]) \to L^2([0,N])$ as

$$\begin{aligned}\mathcal{J} f &= (\chi \otimes \chi) f \\ &= \langle f, \chi \rangle_{L^2([0,N])} \chi \\ &= \mathcal{G}\mathcal{H} f.\end{aligned} \qquad (17)$$

The eigenvalues and eigenvectors of $\mathcal{F} + \frac{1}{2}\mathcal{J}$ can be analytically determined under the condition of χ.

Theorem 2. *Let $\chi \in C^1([0,N])$ satisfy $\chi(x) \neq 0$ for any $x \in [0,N]$. The eigenvalues λ_k and eigenvectors f_k of $\mathcal{F} + \frac{1}{2}\mathcal{J}$ are given as follows*

$$\begin{cases} \lambda_k = \dfrac{1}{(2k+1)i\pi} \displaystyle\int_0^N |\chi(\nu)|^2 d\nu \\ f_k(x) = E\chi(x) \exp\left(-(2k+1)i\pi \dfrac{\int_0^x |\chi(\nu)|^2 d\nu}{\int_0^N |\chi(\nu)|^2 d\nu} \right) \end{cases} \quad (k \in \mathbb{Z}) \qquad (18)$$

where $E \in \mathbb{C}$.

Therefore, $\mathcal{F}$ can be expressed as follows by eigenvalue expansion of the operator.

$$\begin{aligned}\mathcal{F} &= (\mathcal{F} + \frac{1}{2}\mathcal{J}) - \frac{1}{2}\mathcal{J} \\ &= \sum_{k\in\mathbb{Z}} \lambda_k e_k \otimes e_k - \frac{1}{2}\mathcal{G}\mathcal{H}\end{aligned} \qquad (19)$$

where $e_k = f_k/\|f_k\|$. This is a generalization of the DPLR representation in finite dimensions to infinite dimensions.

As a property of the eigenvector, we see that the norm can be expressed using χ and also forms an orthonormal basis of $L^2([0,N])$.

Theorem 3. *Let $\chi \in C^1([0,N])$ satisfy $\chi(x) \neq 0$ for any $x \in [0,N]$. Then,*

$$f_k(x) = \chi(x) \exp\left(-(2k+1)i\pi \frac{\int_0^x |\chi(\nu)|^2 d\nu}{\int_0^N |\chi(\nu)|^2 d\nu} \right) \quad (k \in \mathbb{Z}) \qquad (20)$$

are orthonormal basis of $L^2([0,N])$ and it satisfies

$$\|f_k\|_{L^2([0,N])} = \|\chi\|_{L^2([0,N])}. \qquad (21)$$

Remark 3.1. The λ_k depends only on the norm of χ. Thus, note that changing a function χ that does not pass through the origin can be expressed simply by changing the value of the norm.

3.3. An infinite Dimensional LSSL with Infinite Dimensional HiPPO

Consider solving the equation below formally for infinite dimensional LSSL

$$\frac{\partial h}{\partial t}(t) = (\mathcal{F} - \mathcal{I})h(t) + \mathcal{G}u(t), \tag{22a}$$

$$h(0)(x) = 0 \tag{22b}$$

with the following Infinite Dimensional HiPPO.

We assume that the time derivative and the infinite sum commute, that is,

$$\frac{\partial h}{\partial t}(t) = \sum_{k\in\mathbb{Z}} \frac{\partial c_k(t)}{\partial t} f_k \tag{23}$$

holds when h is represented as

$$h(t) = \sum_{k\in\mathbb{Z}} c_k(t) f_k \tag{24}$$

by the orthogonal basis $\{f_k\}_{k\in\mathbb{Z}}$.

Lemma 3.2.

$$\frac{1}{2}\langle f_k, \chi\rangle_{L^2([0,N])} = \lambda_k, \tag{25}$$

$$\chi = -\frac{2}{\|\chi\|^2}\sum_{k\in\mathbb{Z}} \lambda_k f_k. \tag{26}$$

Using this lemma, it holds that

$$\begin{aligned}\frac{\partial c_k(t)}{\partial t} =& (\lambda_k - 1)c_k(t) + \left(\sum_{m\in\mathbb{Z}} \lambda_m c_m(t)\right)\frac{2}{\|\chi\|^2}\lambda_k \\ &- u(t)\frac{2}{\|\chi\|^2}\lambda_k\end{aligned} \tag{27}$$

as coefficients of formal solutions by Fourier series expansion (see appendix B). Consider the truncated equation:

$$\begin{aligned}\frac{\partial c_k(t)}{\partial t} =& (\lambda_k - 1)c_k(t) + \left(\sum_{m=-M}^{M-1} \lambda_m c_m(t)\right)\frac{2}{\|\chi\|^2}\lambda_k \\ &- u(t)\frac{2}{\|\chi\|^2}\lambda_k \ (k = -M, ..., M-1).\end{aligned} \tag{28}$$

Note that we are left with two terms from the one with the largest absolute value.

If we represent this using a matrix, we can write

$$\frac{\partial c}{\partial t}(t) = (\Lambda - PQ^*)c(t) + Bu(t) \tag{29}$$

$$\Lambda \in \mathbb{C}^{M\times M}, P, Q, B \in \mathbb{C}^{M\times 1} \tag{30}$$

$$\Lambda_{ij} = \lambda_i\delta_{ij},\ P_{i1} = Q_{i1} = \sqrt{2}\frac{\lambda_i}{\|\chi\|},\ B_{i1} = 2\frac{\lambda_i}{\|\chi\|^2}. \tag{31}$$

This is a finite dimensional LSSL using a matrix with a DPLR representation, which can be handled by the S4 model. In particular, this DPLR representation is explicitly obtained. In the next section, we show some experiments with the S4 model using this matrix.

4. Experiments

We performed experiments of the learning task of Sequential MNIST with the S4 model using the above Matrices Λ, P, Q, B.

4.1. Sequential MNIST

MNIST is a dataset for classifying handwritten numbers. Sequential MNIST is a dataset where an image dataset is serialized and treated as a time-series dataset. We used this dataset.

4.2. Details

We used an architecture with the model's internal dimensionality set to 64. For the encoder layer, we used a linear layer. In the middle layers, we used batch normalization, an S4 layer, GELU activation, skip connections, dropout (p=0.1), and GLU activation. The decoder layer included average pooling along the time-series direction and a linear layer. We used 2 layers for the middle layers. As the optimizer, we used Adam with a learning rate of 0.001. For the learning rate scheduler, we used a Cosine with Warmup Scheduler, with a warmup step of 1200 steps. In the S4 layer, we treated A, B, and the time-direction step size Δt as non-trainable hyperparameters and we used $2M = N = 64$ as the state space dimension. We used an NVIDIA GeForce RTX 3060 Laptop GPU for computation and PyTorch as the library.

4.3. Result

The learning curves for training loss, validation loss, and accuracies are shown in Figures 1, 2, and 3, respectively. The final results of the training are summarized in Table 1. It can be seen that the performance of the proposed model is comparable to that of the original model.

Table 1: Results of Training

	Accuracy	Train Loss	Eval Loss
original	0.9930	0.0088	0.0229
$\|\chi\|$=2	0.9904	0.0104	0.0302
$\|\chi\|$=8	0.9917	0.0094	0.0292
$\|\chi\|$=32	0.9777	0.0695	0.0749
$\|\chi\|$=128	0.8396	0.4686	0.4663
$\|\chi\|$=512	0.6777	0.8674	0.8880

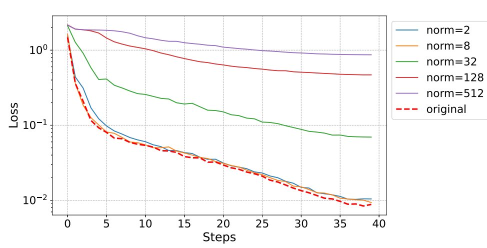

Figure 1: Learning Curve about Training Losses

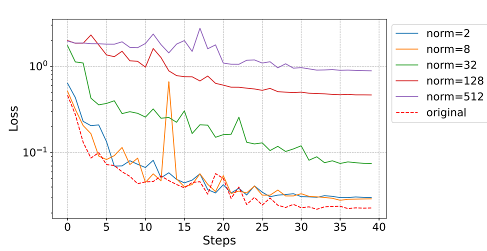

Figure 2: Learning Curve about Validation Losses

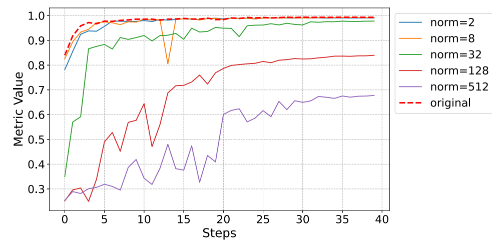

Figure 3: Learning Curve about Accuracies

5. Conclusion

From the viewpoint of partial differential equations of LSSL using HiPPO matrices, the infinite dimensional operator “Infinite dimensional HiPPO” corresponding to HiPPO matrices is discussed, and in particular, it is shown that its eigenvalues and eigenvectors can be obtained analytically. We also proposed a matrix with explicit DPLR representation derived from the infinite dimensional LSSL using Infinite Dimensional HiPPO. Numerical experiments with the S4 model showed that it can be learned in the same way as the HiPPO matrix.

Acknowledgments

This work is supported by JST CREST Grant Number JPMJCR1914 and JPMJCR24Q5, JST ASPIRE JPMJAP2329.

References

[1] Albert Gu, Karan Goel, and Christopher Re. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations*, 2022.

[2] Albert Gu, Isys Johnson, Karan Goel, Khaled Kamal Saab, Tri Dao, Atri Rudra, and Christopher Re. Combining recurrent, convolutional, and continuous-time models with linear state space layers. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021.

[3] Atsushi Takabatake, Baige Xu, and Takaharu Yaguchi. Hyperbolic-PDE-based neural network architecture. In *The 2024 International Symposium on Nonlinear Theory and Its Applications*, to appear.

[4] Atsushi Takabatake and Takaharu Yaguchi. *Hyperbolic partial differential equations generalize LSSL with the HiPPO matrix.* submitted.

A. Proofs of Theorems

Proof of Theorem 2. Solve

$$\begin{aligned}[(\mathcal{F}+\frac{1}{2}\mathcal{J})f](x) =& -\frac{1}{2}\chi(x)\int_0^x f(\xi)\overline{\chi(\xi)}d\xi \\ &+\frac{1}{2}\chi(x)\int_x^N f(\xi)\overline{\chi(\xi)}d\xi \\ =&\lambda f(x)\end{aligned} \tag{32}$$

with $\lambda \neq 0$. If this equation is satisfied, note that f is differentiable in $[0, N]$, and from $\chi(x) \neq 0$, dividing both sides by χ and differentiating, we get

$$\begin{aligned}-f(x)\overline{\chi(x)} &= \lambda\frac{\partial}{\partial x}\left(\frac{f(x)}{\chi(x)}\right) \\ &= \lambda\frac{f'(x)\chi(x)-f(x)\chi'(x)}{(\chi(x))^2}.\end{aligned} \tag{33}$$

By deformation,

$$\lambda f'(x)\chi(x) = \lambda f(x)\chi'(x) - f(x)\overline{\chi(x)}\,(\chi(x))^2 \tag{34}$$

and

$$f'(x) = f(x)\frac{\chi'(x)}{\chi(x)} - \frac{1}{\lambda}f(x)\overline{\chi(x)}\chi(x) \tag{35}$$

hold. Solving this differential equation, we can write f as

$$f(x) = C\chi(x)\exp\left(-\frac{1}{\lambda}\int_0^x |\chi(\nu)|^2 d\nu\right) \tag{36}$$

Substituting into equation (32), we get

$$\begin{aligned}&-\frac{C}{2}\chi(x)\int_0^x |\chi(\xi)|^2\exp\left(-\frac{1}{\lambda}\int_0^x |\chi(\nu)|^2 d\nu\right)d\xi \\ &+\frac{C}{2}\chi(x)\int_x^N |\chi(\xi)|^2\exp\left(-\frac{1}{\lambda}\int_0^x |\chi(\nu)|^2 d\nu\right)d\xi \\ &=\lambda C\chi(x)\exp\left(-\frac{1}{\lambda}\int_0^x |\chi(\nu)|^2 d\nu\right)\end{aligned} \tag{37}$$

Here, it hold that

$$\begin{aligned}&-\int_0^x |\chi(\xi)|^2\exp\left(-\frac{1}{\lambda}\int_0^\xi |\chi(\nu)|^2 d\nu\right)d\xi \\ &+\int_x^N |\chi(\xi)|^2\exp\left(-\frac{1}{\lambda}\int_0^\xi |\chi(\nu)|^2 d\nu\right)d\xi \\ =&\left[\lambda\exp\left(-\frac{1}{\lambda}\int_0^\xi |\chi(\nu)|^2 d\nu\right)\right]_{\xi=0}^{\xi=x} \\ &-\left[\lambda\exp\left(-\frac{1}{\lambda}\int_0^\xi |\chi(\nu)|^2 d\nu\right)\right]_{\xi=x}^{\xi=N} \\ =&2\lambda\exp\left(-\frac{1}{\lambda}\int_0^x |\chi(\nu)|^2 d\nu\right) \\ &-\lambda\exp\left(-\frac{1}{\lambda}\int_0^N |\chi(\nu)|^2 d\nu\right)-\lambda,\end{aligned} \tag{38}$$

hence, by substituting into equation (37) and transforming the equation, we get

$$\exp\left(-\frac{1}{\lambda}\int_0^N |\chi(\nu)|^2 d\nu\right) = -1 \tag{39}$$

Thus, using $k' \in \mathbb{Z}$, eigenvalues is written by

$$-\frac{1}{\lambda}\int_0^N |\chi(\nu)|^2 d\nu = (2k'+1)i\pi, \tag{40}$$

namely

$$\lambda = \frac{1}{-(2k'+1)i\pi}\int_0^N |\chi(\nu)|^2 d\nu. \tag{41}$$

Replacing $k' = -k-1$, we obtain

$$\lambda = \frac{1}{(2k+1)i\pi}\int_0^N |\chi(\nu)|^2 d\nu. \tag{42}$$

Substituting this into (36) yields the eigenvectors.

When $\lambda = 0$, dividing both sides of

$$\begin{aligned}[(\mathcal{F}+\frac{1}{2}\mathcal{J})f](x) =& -\frac{1}{2}\chi(x)\int_0^x f(\xi)\overline{\chi(\xi)}d\xi \\ &+\frac{1}{2}\chi(x)\int_x^N f(\xi)\overline{\chi(\xi)}d\xi \\ =&0\end{aligned} \tag{43}$$

by χ and differentiating results in

$$f(\xi)\overline{\chi(\xi)} = 0, \tag{44}$$

which implies $f = 0$. Therefore, λ cannot be an eigenvalue.

□

Proof of Theorem 3. From the proof of Theorem 2, if $\lambda = 0$, then $f = 0$. That is namely

$$\mathrm{Ker}\left(\mathcal{F} + \frac{1}{2}\mathcal{J}\right) = \{0\}. \tag{45}$$

Therefore,

$$\begin{aligned} L^2([0,N]) &= \mathrm{Ker}\left(\mathcal{F} + \frac{1}{2}\mathcal{J}\right) \oplus \overline{\mathrm{Im}\left(\mathcal{F} + \frac{1}{2}\mathcal{J}\right)} \\ &= \overline{\mathrm{Im}\left(\mathcal{F} + \frac{1}{2}\mathcal{J}\right)} \end{aligned} \tag{46}$$

holds. On the other hand, since elements of $\{f_k\}_{k\in\mathbb{N}}$ are eigenvectors corresponding to a different eigenvalue, it is orthogonal basis for $\overline{\mathrm{Im}\left(\mathcal{F} + \frac{1}{2}\mathcal{J}\right)}$. Thus $\{f_k\}_{k\in\mathbb{N}}$ is an orthogonal basis of $L^2([0,N])$. For norm, it follows from

$$\|f_k\|^2 = \int_0^N f_k(x)\overline{f_k(x)}dx = \int_0^N \chi(x)\overline{\chi(x)}dx = \|\chi\|^2. \tag{47}$$

□

proof of Lemma 3.2.

$$\begin{aligned} &\frac{1}{2}\langle f_k, \chi\rangle_{L^2([0,N])} \\ =&\frac{1}{2}\int_0^N f_k(\xi)\overline{\chi(\xi)}d\xi \\ =&\frac{1}{2}\int_0^N \chi(\xi)\exp\left(-(2k+1)i\pi\frac{\int_0^\xi |\chi(\nu)|^2 d\nu}{\int_0^N |\chi(\nu)|^2 d\nu}\right)\overline{\chi(\xi)}d\xi \\ =&\frac{1}{2}\int_0^N |\chi(\xi)|^2\exp\left(-(2k+1)i\pi\frac{\int_0^\xi |\chi(\nu)|^2 d\nu}{\int_0^N |\chi(\nu)|^2 d\nu}\right)d\xi \\ =&-\frac{1}{2}\cdot\frac{1}{(2k+1)i\pi}\int_0^N |\chi(\nu)|^2 d\nu \\ &\cdot\left[\exp\left(-(2k+1)i\pi\frac{\int_0^\xi |\chi(\nu)|^2 d\nu}{\int_0^N |\chi(\nu)|^2 d\nu}\right)\right]_0^N \\ =&\frac{1}{(2k+1)i\pi}\int_0^N |\chi(\nu)|^2 d\nu \\ =&\lambda_k \end{aligned} \tag{48}$$

Because λ_k is pure imaginary,

$$\begin{aligned} \chi &= \sum_{k\in\mathbb{N}}\langle\chi, \frac{f_k}{\|f_k\|}\rangle\frac{f_k}{\|f_k\|} \\ &= \sum_{k\in\mathbb{N}}\frac{1}{\|f_k\|^2}\langle\chi, f_k\rangle f_k \\ &= \frac{2}{\|\chi\|^2}\sum_{k\in\mathbb{N}}\overline{\lambda_k}f_k \\ &= -\frac{2}{\|\chi\|^2}\sum_{k\in\mathbb{N}}\lambda_k f_k \end{aligned} \tag{49}$$

holds. □

B. Formal Solution by Fourier Expansion

Since

$$\begin{aligned}\mathcal{F}h(t) &= \left(\mathcal{F} + \frac{1}{2}\mathcal{J}\right) h(t) - \frac{1}{2}\mathcal{J}h(t) \\ &= \left(\mathcal{F} + \frac{1}{2}\mathcal{J}\right) h(t) - \frac{1}{2}\langle h(t), \chi\rangle_{L^2([0,N])}\chi \\ &= \sum_{k\in\mathbb{Z}} \lambda_k c_k(t) f_k - \sum_{k\in\mathbb{Z}} c_k(t) \cdot \frac{1}{2}\langle f_k, \chi\rangle_{L^2([0,N])}\chi \\ &= \sum_{k\in\mathbb{Z}} \lambda_k c_k(t) f_k - \sum_{k\in\mathbb{Z}} \lambda_k c_k(t)\chi\end{aligned} \tag{50}$$

and

$$\begin{aligned}&(\mathcal{F} - \mathcal{I})h(t) + \mathcal{G}u(t) \\ =&\sum_{k\in\mathbb{Z}} \lambda_k c_k(t) f_k - \left(\sum_{m\in\mathbb{Z}} \lambda_m c_m(t)\right)\chi \\ &- \sum_{k\in\mathbb{Z}} c_k(t) f_k - u(t)\frac{2}{\|\chi\|^2}\sum_{k\in\mathbb{Z}} \lambda_k f_k \\ =&\sum_{k\in\mathbb{Z}} \lambda_k c_k(t) f_k + \left(\sum_{m\in\mathbb{Z}} \lambda_m c_m(t)\right)\frac{2}{\|\chi\|^2}\sum_{k\in\mathbb{Z}} \lambda_k f_k \\ &- \sum_{k\in\mathbb{Z}} c_k(t) f_k - u(t)\frac{2}{\|\chi\|^2}\sum_{k\in\mathbb{Z}} \lambda_k f_k\end{aligned} \tag{51}$$

holds, the differential equation is written by

$$\begin{aligned}&\sum_{k\in\mathbb{Z}} \frac{\partial c_k(t)}{\partial t} f_k \\ =&\sum_{k\in\mathbb{Z}} \lambda_k c_k(t) f_k + \left(\sum_{m\in\mathbb{Z}} \lambda_m c_m(t)\right)\frac{2}{\|\chi\|^2}\sum_{k\in\mathbb{Z}} \lambda_k f_k \\ &- \sum_{k\in\mathbb{Z}} c_k(t) f_k - u(t)\frac{2}{\|\chi\|^2}\sum_{k\in\mathbb{Z}} \lambda_k f_k \\ =&\sum_{k\in\mathbb{Z}} (\lambda_k - 1)c_k(t) f_k \\ &+ \left(\sum_{m\in\mathbb{Z}} \lambda_m c_m(t)\right)\frac{2}{\|\chi\|^2} \\ &- u(t)\frac{2}{\|\chi\|^2}\sum_{k\in\mathbb{Z}} \lambda_k f_k.\end{aligned} \tag{52}$$

Viewing for each component k,

$$\begin{aligned}\frac{\partial c_k(t)}{\partial t} =&(\lambda_k - 1)c_k(t) + \left(\sum_{m\in\mathbb{Z}} \lambda_m c_m(t)\right)\frac{2}{\|\chi\|^2}\lambda_k \\ &- u(t)\frac{2}{\|\chi\|^2}\lambda_k\end{aligned} \tag{53}$$

is derived.