



universität
wien

MASTERARBEIT / MASTER'S THESIS

Titel der Masterarbeit / Title of the Master's Thesis
„Extrapolation of Quantum Time Series“

verfasst von / submitted by
Yoonjeong Shin

angestrebter akademischer Grad / in partial fulfilment of the requirements for the degree of
Master of Science (MSc)

Wien, 2023 / Vienna, 2023

Studienkennzahl lt. Studienblatt /
degree programme code as it appears on
the student record sheet:

UA 066 876

Studienrichtung lt. Studienblatt /
degree programme as it appears on
the student record sheet:

Masterstudium Physics

Betreut von / Supervisor:

Univ. Prof. Dr. Časlav Brukner

Acknowledgments

I sincerely appreciate my supervisors Miguel Navascués, Mirjam Weilenmann and Andrew James Philip Garner for sparing their valuable time for me in the midst of their busy life, giving me constructive guidance throughout this project and bearing with me. I am also indebted to Prof. Brukner, administration staffs at IQOQI and SSC Physics at the University of Vienna for their favor.

I gratefully acknowledge Dr. Exl for sparking my interest in optimization and cheering me up. Thanks are also due to Prof. Gutjahr and Prof. Hermann for imparting their knowledge of optimization to students at the University of Vienna.

Last but not least, I would like to express my sincere gratitude towards my parents and my little sister.

Kurzfassung

Eines der Hauptziele der Physik ist es, anhand der aktuell verfügbaren Informationen vorherzusagen, wie eine physikalische Größe von Interesse in der Zukunft aussehen wird. In diesem Sinne steht die Extrapolation im Zusammenhang mit Physik. Im Bereich der numerischen Analyse bedeutet Extrapolation eine Methode, ein gegebener Datensatz zu erweitern und einen Wert über den Bereich hinaus zu schätzen. In dieser Studie beschränken wir unseren Fokus auf *Hilbert-Schmidt* (HS)-Observablen und versuchen, deren Zeitmittelwerte zu extrapolieren. Zu diesem Zweck entwickeln wir zwei mögliche Szenarien und zwei „HS-Extrapolationsfunktionen“, die auf Superoszillationen bzw. McLaurin-Expansion basieren. Während wir versuchen, den mit der Extrapolation verbundenen Fehler zu minimieren, stellen wir fest, dass beide Szenarien auf eine konvexe Optimierung über Extrapolationsfunktionen mit minimaler l_1 -Norm hinauslaufen. Zur Vereinfachung wird ein funktionaler Ansatz in Form einer Reihe gewählt und das Optimierungsproblem hinsichtlich seines Koeffizientenvektors mithilfe der Softwarepakete Mosek und CVX gelöst. Bemerkenswerterweise prägen „spärliche Koeffizienten“ die optimale Extrapolationsfunktionen, d. h. die Funktionen haben nur wenige Komponenten ungleich Null. In diesem Zusammenhang untersuchen wir zunächst, wie sich ihre Indizes mit dem Schätzfehler δ ändern, was uns eine Gruppierung ermöglicht. Gleichzeitig schließen wir aus der Tatsache, dass die Anzahl der dünnen Koeffizienten zunimmt, wenn δ abnimmt, dass der Koeffizientenvektor größtenteils Komponenten ungleich Null aufweist, wenn δ sich Null annähert. Andererseits zeigen lineare Anpassungsfunktionen an Werte jeder Gruppe, dass ihre Werte mit hoher Genauigkeit vorhergesagt werden können. Zuletzt erweitern wir den Ansatz auf einen zukünftigen Zeitpunkt τ , indem wir die gleichen Schritte in Bezug auf τ wiederholen. Während dieses Prozesses wird eine neue Indexgruppe entdeckt, die unser Vertrauen in die Schlussfolgerung stärkt. Anschließend wenden wir uns dem Fehlermodell zu, das angibt, wie zuverlässig jeder Punkt in der Zeitreihe ist. Um die Entsprechung zwischen Extrapolationsfunktion und Fehlermodellen herauszufinden, formulieren wir das ursprüngliche Optimierungsproblem basierend auf seiner Dualität neu und betrachten drei Extrapolationsfunktionen: Lagrange-Polynome und die beiden oben erwähnten HS Extrapolationsfunktionen. Das umformulierte Problem nimmt als Zielfunktion die Dualitätslücke an, die als Indikator für die Optimalität einer gegebenen Funktion mit dem Fehlermodell dient. Dabei vergleichen wir die Optimalität der Kandidatenfunktionen anhand ihrer optimalen Werte. Es stellt sich heraus, dass keine der betrachteten Funktionen für irgendein Fehlermodell optimal ist. Sie sind jedoch alle nahezu optimal für das Null-Fehler-Modell.

Abstract

It is one of the primary goals of physics to predict what a physical quantity of interest is going to be like in the future based on currently available information. In this sense, *extrapolation* is along the lines of physics. In the field of numerical analysis, extrapolation refers to a method to extend a given set of data points and estimate a value beyond the range. In this study, narrowing down our focus to *Hilbert-Schmidt* (HS) *observables*, we try to extrapolate their time averages. To this end, we come up with two possible scenarios and two 'HS extrapolation functions', based on *superoscillations* and *McLaurin expansion*, respectively. While trying to minimize the error associated with the extrapolation, we find that both scenarios boil down to a *convex optimization* over extrapolation functions with minimum ℓ_1 norm. For the sake of simplicity, a functional ansatz in the form of a series is adopted and the optimization problem is solved with respect to its coefficient vector by utilizing the software packages named *Mosek* and *CVX*. Remarkably, optimal extrapolation functions feature 'sparse coefficients', namely, only few non-zero components. In this regard, we first study how their indices change with the estimation error δ , which allows us to group them. At the same time, from the fact that the number of sparse coefficients increases as δ decreases, we deduce that the coefficient vector ends up with mostly non-zero components as δ approximates zero. On the other hand, linear fit functions to values of each group reveal that their values can be predicted with high accuracy. Lastly, we extend the ansatz to a future time point τ , by repeating the same steps with respect to τ . During this process, a new index group is discovered, which strengthens our confidence in the deduction. Afterwards, we turn our attention to the error model, which indicates how reliable each point in the time series is. In order to figure out the correspondence between extrapolation function and error models, we reformulate the original optimization problem based on its *duality* and consider three extrapolation functions: *Lagrange polynomials* and the two HS extrapolation functions mentioned above. The recast problem takes as its objective function the *duality gap*, which serves as an indicator of the optimality of a given function with the error model. We thereby compare the optimality of the candidate functions based on their optimal values. It turns out that none of the considered functions is optimal for any error model. However, they are all close to optimal for the zero-error model.

Keywords : Numerical analysis, Extrapolation, Matrix state approximation, Hilbert-Schmidt operators/observables, Convex optimization, CVX, Mosek.

Contents

Acknowledgements	i
Kurzfassung	ii
Abstract	iii
I . Introduction	1
1.1 Statement of the problem	1
1.2 Extrapolation method for Hilbert-Schmidt observables	3
1.3 Examples of HS extrapolation function	4
1.4 From extrapolation to optimization	8
II. Optimization	13
2.1 Fundamentals	13
2.2 Duality	20
2.3 CVX and Numerical solvers	25
III. Hilbert-Schmidt extrapolation function	27
3.1 Optimization problem.....	27
3.2 Results.....	28
IV. Error models	39
4.1 Optimization problem	39
4.2 Potentially optimal extrapolation functions	41
4.3 Results	46
V. Conclusion	51

Bibliography	53
List of Figures	57
List of Tables	61
Appendix A Justification of the equation (1.4.5)	63
Appendix B Fit functions of delta and tau	65
Appendix C Ideas from duality	69
Appendix D Error models associated with Section 4.3	73
Appendix E Ideas for further extrapolation functions	79

Chapter I

Introduction

1.1 Statement of the problem*

Consider a quantum system represented by Hilbert space \mathcal{H} and an associated Hamiltonian $H \in [0, E_{max}]$ and let A denote a self-adjoint operator on the Hilbert space \mathcal{H} . We assume that we can estimate up to a point the averages with respect to an arbitrary state $|\psi_0\rangle \in \mathcal{H}$

$$a(t) := \langle \psi_0 | e^{iHt} A e^{-iHt} | \psi_0 \rangle \quad (1.1.1)$$

for any time $t \in [0, T]$. We wish to *extrapolate* the available data (1.1.1) and provide an estimator for $a(\tau)$ where $\tau \notin [0, T]$. *Extrapolation* in a nutshell is a method to estimate a physical quantity of interest beyond the observable range given a set of available data points. As for the situation described above, the observable associated with A is the quantity of interest, the time interval $[0, T]$ is the observable range, and available data is $a(t)$ as defined as (1.1.1). Then extrapolation of the observable of A is to estimate $a(\tau)$ where τ is a future point outside the given time interval, that is, $\tau > T$, given a set of data points $(t_i, a(t_i))$ with $t_i \in [0, T]$ and $i \in \mathbb{Z}_{\geq 0}$. Extrapolation is often compared with *interpolation* to estimate a quantity between available data points, that is, $a(t')$ where $t_i < t' < t_{i+1}$. One of the most widely used interpolation methods, the so-called *Lagrange polynomial*, will be introduced in Chapter 4 and used as an extrapolation function for the situation under our consideration. There can be many different scenarios which realize our assumption to estimate $a(t)$; here we conceptually describe two possibilities:

Scenario (a) Imagine carrying out a numerical computation that provides us with an estimation $\tilde{a}(t)$ for $a(t)$, with $|\tilde{a}(t) - a(t)| \leq \epsilon(t)$. Here $\epsilon(t)$ denotes the *error model* which gives an upper bound on the positive difference between $\tilde{a}(t)$ and $a(t)$ at time t . This scenario corresponds with simulating the evolution of a 1-D equation system via matrix product state (MPS) approximations [1].

* For ease of notation, consider the Plank constant \hbar to be absorbed by the Hamiltonian H , that is, $H/\hbar \rightarrow H$. Accordingly, when it comes to its observable E , $E/\hbar \rightarrow E$.

Chapter I . introduction

Given a matrix product state approximation $|\tilde{\psi}(t)\rangle$ for the state of the system at time t , its time evolution in δt can be approximated by the following equation

$$|\tilde{\psi}(t + \delta t)\rangle := \arg \max_{\psi \in \mathcal{M}(D)} |\langle \psi | I - iH\delta t | \tilde{\psi}(t)\rangle|^2 \quad (1.1.2)$$

where $\mathcal{M}(D)$ denotes the set of matrix product states of bond dimension of D , or other family of tensor network states [1]. The infinitesimal changes in time δt guarantees the goodness of the estimation to some extent by restricting the deviation of $\tilde{a}(t)$ from $a(t)$ to $\epsilon(t)$ at each time step. Although we wish to approximate $|\tilde{\psi}(\tau)\rangle$ and thereby $\tilde{a}(\tau)$ to estimate $a(\tau)$, the MPS approximations can break before we reach the target point τ . In this case, $\epsilon(t)$ is proportional to $|A|$.

Proof. Define $\hat{\epsilon}(t)$ as the error between $\text{proj} |\tilde{\psi}(t)\rangle$ and $\text{proj} |\psi(t)\rangle$. Here proj stands for the projection operator by which the states are projected into the set of MPS with bond dimension D [2]. Given the definition of $\hat{\epsilon}(t)$, it follows that, for all t ,

$$|\text{proj} |\tilde{\psi}(t)\rangle - \text{proj} |\psi(t)\rangle|_1 \leq \hat{\epsilon}(t) \quad (1.1.3)$$

and

$$\hat{\epsilon}(t) \leq 2 \quad (1.1.4)$$

From these considerations, we have

$$\begin{aligned} |\tilde{a}(t) - a(t)| &= |\text{tr}\{A(\text{proj} |\tilde{\psi}(t)\rangle - \text{proj} |\psi(t)\rangle)\}| \\ &\leq |A| |\text{proj} |\tilde{\psi}(t)\rangle - \text{proj} |\psi(t)\rangle|_1 \\ &\leq |A| \hat{\epsilon}(t) \leq 2|A| \end{aligned} \quad (1.1.5)$$

□

Scenario (b) Consider an experiment which starts with the initial state $|\psi_0\rangle$. Let it evolve for some time $t \in [0, T]$ and measure the observable concerning A . Let us denote corresponding results as a_t . We would better not allow the system to evolve for τ at once so as to prevent disintegration of the system due to its interaction with environment. By repeating this experiment several times and averaging out all values of a_t measured, we can obtain a statistical estimator for $a(t)$. Employing an unbiased probability distribution, in other words, a probability distribution with a low variance is accordingly considered as a natural choice to improve the goodness of the estimation.

These two possible scenarios to come up with an estimator for $a(\tau)$ with $\tau \notin [0, T]$ will be revisited and examined in detail in Section 1.4 with the mathematical formulation on *Hilbert-Schmidt (HS) observables*, which will be explored in the next section.

1.2 Extrapolation method for Hilbert-Schmidt observables

Extrapolation methods should be chosen with caution depending on properties of the operator A . For instance, whether the operator A is HS needs to be considered; this is the type of operators our study relates to.

Definition 1.2.1 (HS operator) *An operator A is called Hilbert-Schmidt (HS) if*

$$|A|_2 = \sqrt{\text{tr}(A^2)} \quad (1.2.1)$$

Let $|A\rangle$ be the vector form of the operator $A = \sum_{jk} A_{jk} |j\rangle\langle k|$, i.e., $|A\rangle = \sum_{jk} A_{jk} |j\rangle|k\rangle$, where $\{|j\rangle\}_j$ is a real orthonormal basis. Note that $|A|_2^2 = \langle A|A\rangle$. Let E_j and $|\phi_j\rangle$ respectively denote the j -th eigenvalue of H and its corresponding eigenvector. Then $a(t)$ can be expressed as

$$\begin{aligned} a(t) &:= \langle \psi_0 | e^{iHt} A e^{-iHt} | \psi_0 \rangle \\ &= \sum_{jk} A_{jk} \langle \psi_0 | e^{iHt} | j \rangle \langle k | e^{-iHt} | \psi_0 \rangle \\ &= \sum_{jk} A_{jk} \langle k | e^{-iHt} | \psi_0 \rangle \langle \psi_0 | e^{iHt} | j \rangle \\ &= \sum_{jk} A_{jk} \langle k | e^{-iHt} | \psi_0 \rangle \langle j | e^{iHt} | \bar{\psi}_0 \rangle \\ &= \sum_{jk} \bar{A}_{kj} \langle k | \langle j | e^{-iHt} | \psi_0 \rangle e^{iHt} | \bar{\psi}_0 \rangle \\ &= \langle A | (e^{-iHt} \otimes e^{iHt}) | \psi_0 \rangle | \bar{\psi}_0 \rangle \\ &= \langle A | \sum_{jk} e^{-i(E_j - E_k)t} (|\phi_j\rangle\langle\phi_j| \otimes |\bar{\phi}_k\rangle\langle\bar{\phi}_k|) | \psi_0 \rangle | \bar{\psi}_0 \rangle \end{aligned} \quad (1.2.2)$$

For any functions g, h whose domain is $[0, T]$, let us define

$$\langle g, h \rangle := \int_0^T g(t) h(t) dt \quad (1.2.3)$$

Note that this quantity is order-invariant, that is, $\langle g, h \rangle = \langle h, g \rangle$. Suppose that there exists a function $f : [0, T] \rightarrow \mathbb{R}$ such that

$$\Delta(E) := \int_0^T f(t) e^{-iEt} dt - e^{-iEt} =: \langle f, e^{-iEt} \rangle - e^{-iEt} \quad (1.2.4)$$

satisfies

$$|\Delta(E)| \leq \delta, \quad E \in [-E_{max}, E_{max}] \quad (1.2.5)$$

Such a function f will be henceforth referred to as an *HS function*. In order to convince readers of the existence of HS function, spared is the very next section where two examples are given. For an HS extrapolation function f and a finite $|A|_2$ defined as Definition 1.2.1, the quantity $\langle f, a \rangle$ can make a reasonable approximation to $a(\tau)$ as follows:

$$\begin{aligned}
 & |\langle f, a \rangle - a(\tau)| \\
 &= \left| \int_0^T f(t)a(t) dt - a(\tau) \right| \\
 &= \left| \langle A \left\{ \sum_{jk} \int_0^T f(t)e^{-i(E_j-E_k)t} dt |\phi_j\rangle\langle\phi_j| \otimes |\bar{\phi}_k\rangle\langle\bar{\phi}_k| \right\} |\psi_0\rangle |\bar{\psi}_0\rangle - \right. \\
 &\quad \left. \langle A \left\{ \sum_{jk} e^{-i(E_j-E_k)\tau} |\phi_j\rangle\langle\phi_j| \otimes |\bar{\phi}_k\rangle\langle\bar{\phi}_k| \right\} |\psi_0\rangle |\bar{\psi}_0\rangle \right| \\
 &= \left| \langle A \left\{ \sum_{jk} \left[\int_0^T f(t)e^{-i(E_j-E_k)t} dt - e^{-i(E_j-E_k)\tau} \right] |\phi_j\rangle\langle\phi_j| \otimes |\bar{\phi}_k\rangle\langle\bar{\phi}_k| \right\} |\psi_0\rangle |\bar{\psi}_0\rangle \right| \tag{1.2.6} \\
 &= \left| \langle A \left\{ \sum_{jk} \Delta(E_j - E_k) |\phi_j\rangle\langle\phi_j| \otimes |\bar{\phi}_k\rangle\langle\bar{\phi}_k| \right\} |\psi_0\rangle |\bar{\psi}_0\rangle \right| \\
 &\leq |A| \left| \left\{ \sum_{jk} \Delta(E_j - E_k) |\phi_j\rangle\langle\phi_j| \otimes |\bar{\phi}_k\rangle\langle\bar{\phi}_k| \right\} |\psi_0\rangle |\bar{\psi}_0\rangle \right| = |A|_2 |\Delta(E)| \\
 &\leq |A|_2 \delta
 \end{aligned}$$

where the definition (1.2.3) is used to substitute $\langle f, a \rangle$ in the first line, the equation (1.2.2) $a(t)$ and $a(\tau)$ in the second line, the definition (1.2.4) $\Delta(E_j - E_k)$ in the fourth line, and finally the inequality (1.2.5) results in the last line.

1.3 Examples of HS extrapolation function

As previously mentioned, in this section we will introduce two examples of HS extrapolation functions: one inspired by *superoscillations* and the other based on *McLaurin expansions*. Derived and proven to make an HS extrapolation function, that is, proven to satisfy the inequality (1.2.5) in this section, these two functions will be revisited and further elaborated in Chapter 4.

(A) An HS extrapolation function based on superoscillations

Superoscillations [3], discovered by Yakir Aharonov and his collaborators, is a phenomenon in which in terms of Fourier Analysis a bandwidth-limited pulse has a Fourier component which oscillates faster than its bandwidth originally allows [4]. The notion of superoscillating (Fourier) sequence leads us to an HS extrapolation function.

Definition 1.3.1 (Generalized Fourier sequence) [4] A generalized Fourier sequence (or series) is of the form

$$X_N(x, \alpha) := \sum_{j=0}^N F_j(N, \alpha) e^{ik_j(N)x} \quad (1.3.1)$$

where $\alpha \in \mathbb{R}_{\geq 0}$, $N \in \mathbb{N}$, $F_j(N, \alpha)$ and $k_j(N)$ are real functions.

Definition 1.3.2 (Superoscillating Fourier sequence) [4] A generalized Fourier sequence (1.3.1) is said to be superoscillating if

- $k_j(N) < \beta \in \mathbb{R}_{\geq 0} \forall N$ and $j \in \mathbb{N} \cup \{0\}$;
- there exists a compact subset of \mathbb{R} on which $X_N(x, \alpha)$ uniformly converges to $e^{i\mathbf{K}(\alpha)x}$ where $\mathbf{K}(\alpha)$ is a continuous real function such that $|\mathbf{K}(\alpha)| > \beta$.

Consider the following function with respect to $E \in \mathbb{R}$ and $\tau, T \in \mathbb{R}_{\geq 0}$:

$$Y_N(E, \tau, T) := \left(1 - \frac{\tau}{T} + \frac{\tau}{T} e^{-iE\frac{T}{N}}\right)^N \quad (1.3.2)$$

This function can be shown to be superoscillating once *Newton's binomial theorem (or expansion)* is applied to itself.

Theorem 1.3.3 (Newton's binomial) For $N \in \mathbb{Z}_{\geq 0}$, the summation in a particular form can be expanded as

$$(x + y)^N = \sum_{j=0}^N \binom{N}{j} x^{N-j} y^j = \sum_{j=0}^N \binom{N}{j} x^j y^{N-j} \quad (1.3.3)$$

By theorem 1.3.3, the function (1.3.2) can be written in the form of Fourier sequence:

$$Y_N(E, \tau, T) = \sum_{j=0}^N \binom{N}{j} \left(1 - \frac{\tau}{T}\right)^{N-j} \left(\frac{\tau}{T}\right)^j e^{-i\frac{T}{N}jE} \quad (1.3.4)$$

Indeed,

$$\lim_{N \rightarrow \infty} Y_N(E, \tau, T) = e^{-i\tau E} \quad (1.3.5)$$

Having observed the convergence (1.3.5), define quantities $k'_j(N) := -\frac{T}{N}j$ with $j = 0, \dots, N$ and $\mathbf{K}'(\tau) := -\tau$ analogous to $k_j(N)$ and $\mathbf{K}(\alpha)$ in Definition 1.3.2. Then we have

$$\begin{aligned} -T < k'_j(N) < 0 \\ |\mathbf{K}'(\tau)| &= \tau \end{aligned} \quad (1.3.6)$$

Chapter I. introduction

From these, one can easily find the compact set $\beta \in [0, \tau]$ which satisfies the two conditions to be a superoscillating sequence as given in Definition 1.3.2. \square

Furthermore, the limit (1.3.5) together with the sequence (1.3.4) itself implies superoscillations in which a local segment characterized by τ , deviates from the original bandwidth specified by the parameter $\frac{T}{N}j$ which ranges from zero to T with $j = 0, \dots, N$. The limit (1.3.5), which follows from standard computation, can be reversely checked with ease by different definitions of exponential function and the approximation where $N \gg |E| \max(T, \tau)$ for fixed quantities E, T, τ :

$$e^{-i\tau E} \approx \left(1 - \frac{1}{N}i\tau E\right)^N \approx \left(1 - \frac{\tau}{T} + \frac{\tau}{T}e^{-iE\frac{T}{N}}\right)^N =: Y_N(E, \tau, T) \quad (1.3.7)$$

Finally, with respect to E , the series in (1.3.2) is of the form

$$\int_0^T f_S(t) e^{-iEt} dt \quad (1.3.8)$$

where

$$f_S(t) := \sum_{j=0}^N (c_S)_j \delta(t - (t_S)_j) \quad (1.3.9)$$

with

$$\begin{aligned} (c_S)_j &:= \binom{N}{j} \left(1 - \frac{\tau}{T}\right)^{N-j} \left(\frac{\tau}{T}\right)^j \\ (t_S)_j &:= \frac{T}{N}j \end{aligned} \quad (1.3.10)$$

The function (1.3.9) indeed satisfies the inequality (1.2.5), meaning that it is an HS extrapolation function.

(B) An HS extrapolation function based on McLaurin expansion

Taylor expansion (or *Taylor series*), widely used to approximate analytic functions, provides a foundation for the other HS extrapolation function.

Theorem 1.3.4 (Taylor) [5] *Let $G(x)$ have $j + 1$ continuous derivatives on $[a, b]$ for some $j \in \mathbb{Z}_{\geq 0}$. Then, for $t, t_0 \in [a, b]$, the function $G(x)$ can be written as*

$$G(t) = P_j(t) + R_{j+1}(t) \quad (1.3.11)$$

where

$$\begin{aligned} P_j(t) &:= G(t_0) + \frac{G'(t_0)}{1!}(t-t_0) + \dots + \frac{G^{(j)}(t_0)}{j!}(t-t_0)^j \\ R_{j+1}(t) &:= \frac{1}{j!} \int_{t_0}^t G^{(j+1)}(t) (t-t_0)^j dt = \frac{G^{(j+1)}(c)}{(j+1)!} (t-t_0)^{j+1} \end{aligned} \quad (1.3.12)$$

for $c \in [t_0, t]$. $R_{j+1}(t)$ is called *remainder* which means the difference between the function $G(t)$ and the j -th order Taylor polynomial $P_j(t)$, that is, $R_{j+1}(t) = G(t) - P_j(t)$. In other words, the remainder represents the approximation error.

McLaurin expansion (or *McLaurin series*) is a particular case of Taylor expansion with $t_0 = 0$. That is, McLaurin expansion of the function $G(t)$ takes the form

$$G(t) \approx \sum_{j=0}^N \frac{G^{(j)}(t)}{j!} t^j \quad (1.3.13)$$

Here we discarded the remainder $R_{j+1}(t)$ and used the approximately equal sign ' \approx ' instead. Applying the McLaurin expansion (1.3.13) to the function $e^{-iE\tau}$, we have

$$e^{-iE\tau} \approx \sum_{j=0}^N \frac{d^j(e^{-iEt})}{dt^j} \Big|_{t=0} \frac{\tau^j}{j!} \quad (1.3.14)$$

By numerical differentiation [6]

$$\frac{d^j H(t)}{dt^j} = \lim_{h \rightarrow 0} \frac{1}{h^j} \sum_{k=0}^j \binom{j}{k} (-1)^{j+k} H(t+kh) \quad (1.3.15)$$

the right-hand side of the approximation (1.3.14) can be further approximated and expanded as

$$\begin{aligned} \sum_{j=0}^N \frac{d^j(e^{-iEt})}{dt^j} \frac{\tau^j}{j!} &\approx \sum_{j=0}^N \sum_{k=0}^j \frac{\tau^j}{j! h^j} \binom{j}{k} (-1)^{j+k} e^{-iE(kh)} \\ &= \int_0^T f_M(t) e^{-iEt} dt \end{aligned} \quad (1.3.16)$$

where we define $f_M(t)$ as

$$f_M(t) := \sum_{j=0}^N \sum_{k=0}^j \frac{\tau^j}{j! h^j} \binom{j}{k} (-1)^{j+k} \delta(t-kh) \quad (1.3.17)$$

For sufficiently high N and small h , the function $f_M(t)$ satisfies the inequality (1.2.5) and the quantity $\langle f_M, e^{-iEt} \rangle$ converges to the first $N+1$ terms of the McLaurin expansion on τ of the function $e^{-iE\tau}$ as $h \rightarrow 0$.

1.4 From extrapolation to optimization

Having introduced the concept of HS extrapolation function in Section 1.2 and studied examples subsequently, now we are ready to deal with the two scenarios, presented in Section 1.1, in more detail. In the end, we will see that the efficient extrapolation of the averages (1.1.1) boils down to solving an optimization problem over extrapolation functions regardless of extrapolation scenarios.

Scenario (a) With the definition (1.2.3) in mind, note that the following holds:

$$|\langle \tilde{a}, f \rangle - \langle a, f \rangle| \leq \int_0^T |f(t)| |\tilde{a}(t) - a(t)| dt \leq \int_0^T |f(t)| \varepsilon(t) dt \quad (1.4.1)$$

The error model $\varepsilon(t)$ plays an important role throughout this study from bounding $|\tilde{a}(t) - a(t)|$ to controlling time-(in)dependent errors affecting the system. The latter function will be highlighted in Chapter 4. By adding the two inequalities (1.2.6) and (1.4.1), we get

$$|\langle \tilde{a}, f \rangle - a(\tau)| \leq \delta |A|_2 + \int_0^T |f(t)| \varepsilon(t) dt \quad (1.4.2)$$

As for a constant error model $\varepsilon(t) = \varepsilon_0$ for all t , this inequality transforms into

$$|\langle \tilde{a}, f \rangle - a(\tau)| \leq \delta |A|_2 + |f|_1 \varepsilon_0 \quad (1.4.3)$$

with

$$|f|_1 := \int_0^T |f(t)| dt \quad (1.4.4)$$

Scenario (b) One way to estimate $\langle a, f \rangle$ is to choose $t \in [0, T]$ at random according to the distribution

$$\mu(t) dt \equiv \frac{|f(t)|}{|f|_1} dt \quad (1.4.5)$$

Next, we define the random variable $\alpha := a_t \text{sign}(f(t)) |f|_1$ whose value is in $|f|_1 \sigma(A) \cup |f|_1 \sigma(-A)$ where $\sigma(\pm A)$ denote the spectra of $\pm A$. α is an unbiased estimator for $\langle a, f \rangle$. Indeed,

$$\begin{aligned} \langle \alpha \rangle &= \int_0^T \langle \alpha \rangle \mu(t) dt = \int_0^T \langle \alpha_t \rangle \text{sign}(f(t)) |f(t)| dt = \int_0^T a(t) f(t) dt \\ &=: \langle a, f \rangle \end{aligned} \quad (1.4.6)$$

Not only that, but, as shown in Appendix A, α is the unbiased estimator of $\langle a, f \rangle$ with minimum variance. By the equality (1.4.6), the inequality (1.2.6) can be written as

$$|\langle \alpha \rangle - a(\tau)| \leq |A|_2 \delta \quad (1.4.7)$$

However, we do not have direct access to the average $\langle \alpha \rangle$. Rather, we will approximate its value as $\bar{\alpha} := \frac{1}{N} \sum_{i=1}^N \alpha_i$. How close is $\bar{\alpha}$ to $\langle \alpha \rangle$? Chebyshev's inequality, applicable to a wide range of probability distributions, gives us an upper bound.

Theorem 1.4.1 (Chebyshev's inequality) [7] *Consider a probability distribution where X is value of the variable, μ the mean and σ the standard deviation. Then for any $\gamma \in \mathbb{R}_{\geq 0}$,*

$$P(|X - \mu| \geq \gamma \sigma) \leq \frac{1}{\gamma^2} \quad (1.4.8)$$

By Chebyshev's inequality, the probability that the N -sample mean of α significantly diverges from $\langle \alpha \rangle$ is bounded as follows:

$$P(|\bar{\alpha} - \langle \alpha \rangle| \geq \omega) \leq \frac{(\Delta \alpha)^2}{N \omega^2} \leq \frac{|f|_1^2 |A|^2}{N \omega^2} \quad (1.4.9)$$

Now, let us spell out what kind of result we want. We wish that, with (fixed) high probability $1 - p$, with $p \ll 1$, our estimator $\bar{\alpha}$ is close to $a(\tau)$. By the equation above, it follows that

$$\frac{|f|_1 |A|}{\sqrt{Np}} \leq \omega \quad (1.4.10)$$

is enough to guarantee that

$$|\bar{\alpha} - \langle \alpha \rangle| \leq \omega \quad (1.4.11)$$

with probability at least $1 - p$. In that case,

$$|\bar{\alpha} - \langle \alpha \rangle| \leq |A|_2 \delta + \omega \quad (1.4.12)$$

Chapter I . introduction

Hence, to guarantee that our estimator is close to $a(\tau)$ with probability at least $1 - p$, we are interested in minimizing the right-hand side of the equation above.

Recall that, regardless of scenarios, we wish to extrapolate the available data (1.1.1) and provide an estimation of $a(\tau)$ for $\tau \notin [0, T]$. Improving the extrapolation for $a(\tau)$ corresponds with tightening (i.e., minimizing) the upper bounds of the inequalities (1.4.2) and (1.4.12), which leads us to the notion of optimization. Fundamentals of optimization will be accordingly given in the next chapter. The optimization problem associated with scenario (a) then takes the form

$$\begin{aligned} & \underset{f}{\text{minimize}}_f \quad |A|_2 \delta + \int_0^T |f(t)| \varepsilon(t) dt \\ & \text{subject to} \quad |\Delta(E)| \leq \delta, \quad \forall E \in [-E_{max}, E_{max}] \end{aligned} \quad (1.4.13)$$

and, as for a constant error model $\varepsilon(t) = \varepsilon_0$ for all t , the problem transforms into

$$\begin{aligned} & \underset{f}{\text{minimize}}_f \quad |A|_2 \delta + |f|_1 \varepsilon_0 \\ & \text{subject to} \quad |\Delta(E)| \leq \delta, \quad \forall E \in [-E_{max}, E_{max}] \end{aligned} \quad (1.4.14)$$

while the other concerning scenario (b) is of the form

$$\begin{aligned} & \underset{f}{\text{minimize}}_f \quad |A|_2 \delta + \frac{|f|_1 |A|}{\sqrt{Np}} \\ & \text{subject to} \quad |\Delta(E)| \leq \delta, \quad \forall E \in [-E_{max}, E_{max}] \end{aligned} \quad (1.4.15)$$

Although these optimization problems (1.4.14) and (1.4.15) have different objective functions to minimize, due to the fact that both concern the quantity $|f|_1$, of all the extrapolation functions with maximum error δ , we are interested in the ones with minimum $|f|_1$. Moreover, we can practically solve both the two optimization problems at once by dealing with the following optimization problem in a much simpler form than the original ones:

$$\begin{aligned} & \underset{f}{\text{minimize}}_f \quad |f|_1 \\ & \text{subject to} \quad |\Delta(E)| \leq \delta, \quad \forall E \in [-E_{max}, E_{max}] \end{aligned} \quad (1.4.16)$$

Therefore, this optimization problem (1.4.16) makes one of the main subjects of this study; in Chapter 3, a series of trial and error undergone while trying to solve this problem will be presented. The other main subject is the error model $\varepsilon(t)$ as suggested earlier. Multiplied by $f(t)$ as in the inequality (1.4.1), it controls some time-(in)dependent errors which affect the system; whereas a constant error model

corresponds to an unweighted extrapolation function, a time-dependent error model does to the one weighted in a particular way with time. In Chapter 4, we will reformulate the optimization problem (1.4.16) in order to find a correspondence between extrapolation functions and error models. Together with *Lagrange polynomial*, which will be explained in the chapter, the HS extrapolation functions discussed in Section 1.3 will be considered as potential extrapolation functions which match a particular error model. Lastly, a correlation between different types of error models and extrapolation functions will be studied. Until then we consider the error model in the simplest form $\varepsilon(t) = 1$.

Chapter II

Optimization

2.1 Fundamentals*

An optimization problem consists of the following four elements:

- The *vector* $x \in \mathbb{R}^n$ with $n \in \mathbb{N}$;
- The *objective function* $f : \mathbb{R}^n \rightarrow \mathbb{R}$ that takes as input the vector x ;
- The *inequality constraint functions* $g_i : \mathbb{R}^n \rightarrow \mathbb{R}$ where the index $i \in I$, the set of indices of inequality constraints;
- The *equality constraint functions* $h_j : \mathbb{R}^n \rightarrow \mathbb{R}$ where the index $j \in \mathcal{E}$, the set of indices of equality constraints.

While the first two elements are necessary to define an optimization problem, the rest are optional. If an optimization problem is not subject to any constraints, the problem is said to be *unconstrained*, otherwise, *constrained*. In a broad sense, there are two types of optimization: *maximization* and *minimization* (of the objective function). By switching the sign of the objective function, one can easily transform one into the other. A vector x which maximizes or minimizes the objective function is generally called an *optimal solution* (or *optimizer*), which is often marked with an asterisk as x^* . It is indeed not necessarily unique; there can be no solution or many of them. An optimizer is often alternatively referred to as a *maximizer* or a *minimizer* to highlight its role in optimization. With all these underlying concepts, an optimization problem is written as follow; it is conventional to consider minimization.

Definition 2.1.1 (Optimization problem) *An optimization problem takes the standard form*

$$\begin{aligned} & \text{minimize}_x && f(x) \\ & \text{subject to} && g_i(x) \leq 0, \quad i \in \mathcal{E} \\ & && h_j(x) = 0 \quad j \in I \end{aligned} \tag{2.1.1}$$

Moreover, when it comes to a constrained optimization problem, a set of vectors satisfying its constraints is referred to as *feasible region*.

* This section is mostly based on [8] and [9], unless otherwise referenced.

Definition 2.1.2 (Feasible region) The feasible region for the problem (2.1.1) is the set

$$\Omega := \{x \in \mathbb{R}^n \mid g_i(x) \leq 0, \quad i \in \mathcal{E}; \quad h_j(x) = 0, \quad j \in \mathcal{I}\} \quad (2.1.2)$$

An optimization problem is said to be *feasible* if $\Omega \neq \emptyset$, otherwise, *infeasible*.

Furthermore, there are two types of solutions: *global* and *local solutions*. The latter is subdivided into strict and weak ones depending on types of inequalities. The following definitions are written in terms of the minimization problem (2.1.1).

Definition 2.1.3 (Global optimal solution) A vector x^* is called a *global optimal solution* to if $f(x^*) \leq f(x)$ for all $x \in \Omega$.

Definition 2.1.4 (Local optimal solution) A vector x^* is called a *local optimal solution* if there exists a neighborhood N_{x^*} of x^* and $f(x^*) \leq f(x)$ for all $x \in \Omega \cap N_{x^*}$.

Definition 2.1.5 (Strict and weak local optimal solutions) A vector x^* is called a *strict local optimal solution* if there exists a neighborhood N_{x^*} of x^* and $f(x^*) < f(x)$ for all $x \in \Omega \cap N_{x^*}$ and $x \neq x^*$, otherwise, a *weak local optimal solution*.

In addition, optimization problems are called *linear*, *nonlinear*, or *convex*, depending on forms of its objective and constraint functions. A problem whose objective and its constraint functions are all linear is said to be *linear programming*; otherwise, that is, if at least one of them is nonlinear, the associated problem is classified as *nonlinear programming*. Similarly, the problem whose objective and constraint functions are all convex is referred to as *convex programming*. Underlying concepts of convex optimization will be reviewed, followed by those of *conic optimization*, a generalization of linear programming.

Before starting a discussion on convex programming (or *convex optimization*), associated preliminaries will be introduced.

Definition 2.1.6 (Affine set) A set S is said to be *affine* if for any $x_1, x_2 \in S$ and any $\theta \in \mathbb{R}$, we have

$$\theta x_1 + (1 - \theta)x_2 \in S \quad (2.1.3)$$

Definition 2.1.7 (Affine combination) A vector constructed by affine combination of vectors $x_1, x_2, \dots, x_n \in \mathbb{R}^n$ is of the form

$$\sum_{i=1}^n \theta_i x_i \quad (2.1.4)$$

where $\theta_1, \theta_2, \dots, \theta_n \in \mathbb{R}$ satisfying $\sum_{i=1}^n \theta_i = 1$.

Definition 2.1.8 (Affine hull) A affine hull of a set S is the set of all affine combinations of its elements.

$$\text{aff}(S) := \{ \sum_{i=1}^n \theta_i x_i \mid x_i \in S, \theta_i \in \mathbb{R}, \sum_{i=1}^n \theta_i = 1, \} \quad (2.1.5)$$

Definition 2.1.9 (Convex set) A set C is said to be convex if for any $x_1, x_2 \in C$ and any $\varphi \in [0,1]$, we have

$$\varphi x_1 + (1 - \varphi)x_2 \in C \quad (2.1.6)$$

Definition 2.1.10 (Convex combination) A vector constructed by convex combination of points $x_1, x_2, \dots, x_n \in \mathbb{R}^n$ is of the form

$$\sum_{i=1}^n \varphi_i x_i \quad (2.1.7)$$

where $\varphi_1, \varphi_2, \dots, \varphi_n \in [0,1]$ and $\sum_{i=1}^n \varphi_i = 1$.

Definition 2.1.11 (Convex hull) A convex hull of a set C is the set of all convex combinations of its elements.

$$\text{conv}(C) := \{ \sum_{i=1}^n \varphi_i x_i \mid x_i \in C, \sum_{i=1}^n \varphi_i = 1, \varphi_i \in [0,1] \} \quad (2.1.8)$$

These concepts are illustrated in the following three figures;

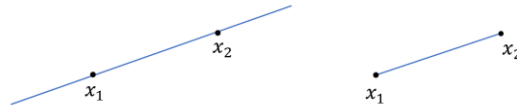


Figure 2-1 Affine (left) and convex (right) sets which have two elements x_1 and x_2 . These can be viewed as affine and convex hulls, respectively.

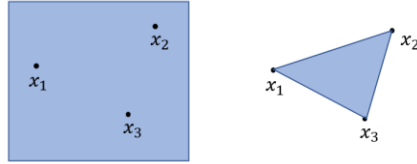


Figure 2-2 Affine (left) and convex (right) sets which have three elements x_1 , x_2 and x_3 . These can be viewed as affine and convex hulls, respectively.



Figure 2-3 Convex (left) and nonconvex (right) sets. The two dots and line segments between them illustrate how to tell convexity of sets geometrically; A set C is convex if a line segment connecting any two points in C lies in C as well, otherwise, nonconvex.

In addition to sets, functions can also be convex. A convex function can be transformed into a concave one by switching its sign, vice versa; and the function which is both convex and concave is commonly called a linear function.

Definition 2.1.12 (convex function) A function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex if $\text{dom } f$, a set of input variables, is a convex set and if, for all $x_1, x_2 \in \text{dom } f$ and $\varphi \in [0,1]$,

$$f(\varphi x_1 + (1 - \varphi)x_2) \leq \varphi f(x_1) + (1 - \varphi)f(x_2) \quad (2.1.9)$$

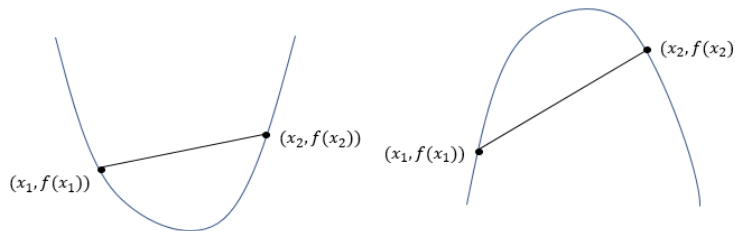


Figure 2-4 Convex (left) and concave (right) functions. From a position of a line segment between any two points on a graph relative to the function one can determine convexity/concavity of a function graphically.

Moreover, replacing the inequality in Definition 2.1.12 with the strict equality leads to the definition of *strict convexity* of a function (Definition 2.1.13) As an extension of this, strong convexity is defined; here we present two most widely used variants of its definition (Definitions 2.1.14 and 2.1.15); for additional variants and proofs of their equivalence, refer to [10].

Definition 2.1.13 (Strict convexity) A function f is strictly convex if the following strict inequality holds for all $x_1, x_2 \in \text{dom } f$ with $x_1 \neq x_2$ and $\varphi' \in (0,1)$:

$$f(\varphi'x_1 + (1 - \varphi')x_2) < \varphi'f(x_1) + (1 - \varphi')f(x_2) \quad (2.1.10)$$

Definition 2.1.14 [10] (Strong convexity var.1) A function f which is continuous over a convex set $C \in \text{dom } f$ with coefficient $\eta' \in \mathbb{R}_{>0}$ is strongly convex if for all $x_1, x_2 \in C$ and $\varphi \in [0,1]$, we have

$$f(\varphi x_1 + (1 - \varphi)x_2) + \frac{\eta'}{2} \varphi(1 - \varphi)|x_1 - x_2|^2 \leq \varphi f(x_1) + (1 - \varphi)f(x_2) \quad (2.1.11)$$

Definition 2.1.15 [10] (Strong convexity var.2) Suppose a function f is twice continuously differentiable over a convex set $C \in \text{dom } f$. The function f is then strongly convex if there exists $\eta \in \mathbb{R}_{\geq 0}$ such that

$$\nabla^2 f(x) - \eta I \geq 0 \quad (2.1.12)$$

By Definitions 2.1.13 and 2.1.14, it is obvious that strong convexity is sufficient for strict convexity.

Additionally, every norm is convex, which can be easily proved as below. Note that both the objective and the inequality constraint functions of the problem (1.4.16) concern norms. Therefore, it follows that the problem is convex optimization. Note that its objective function is convex, yet neither strictly nor strongly convex.

Lemma 2.1.16 (Norms are convex) Every norm $|\cdot|$ is convex

Proof. For any $x_1, x_2 \in \mathbb{R}^n$ and $\varphi \in [0,1]$,

$$\begin{aligned} |\varphi x_1 + (1 - \varphi)x_2| &\leq |\varphi x_1| + |(1 - \varphi)x_2| \\ &= \varphi|x_1| + (1 - \varphi)|x_2| \end{aligned} \quad (2.1.13)$$

The triangular inequality is used in the first line, the homogeneity of norm in the second line. \square

Now let us turn to *convex optimization* (or *convex programming*). Convex optimization is considered to have advantages as they are often easier to analyze and to solve, which is, for example, grounded on the following theorems:

Theorem 2.1.17 If the objective function and feasible region of the optimization problem (2.1.1) are both convex, then any local solution is a global solution.

Proof. Let $x^* \in \Omega$ be a local solution to the optimization problem (2.1.1) and suppose that x^* is not a global solution, meaning that there exists a point $y^* \in \Omega$ such that $f_0(y^*) < f_0(x^*)$. By convexity of the feasible region Ω , we can construct convex combination of x^* and y^* which reads $\varphi x^* + (1 - \varphi)y^* \in \Omega$ for any $\varphi \in [0,1]$. By convexity of f and the inequality from the above assumption $f_0(y^*) < f_0(x^*)$,

Chapter II. Optimization

$$\begin{aligned} f(\varphi x^* + (1 - \varphi)y^*) &\leq \varphi f(x^*) + (1 - \varphi)f(y^*) \\ &< \varphi f(x^*) + (1 - \varphi)f(x^*) = f(x^*) \end{aligned} \quad (2.1.14)$$

This contradicts the local optimality of x^* as φ approaches 1. \square

Theorem 2.1.18 *A convex optimization problem has a unique solution if its objective function is strictly convex.*

Proof. Let f be a strictly convex function and suppose that there were two global optimal solutions $x^*, y^* \in \Omega$, that is, for all $z \in \Omega$,

$$f(x^*) = f(y^*) \leq f(z) \quad (2.1.15)$$

Let us specify a comparison point $z := \frac{x+y}{2}$. Then by convexity of the feasible region Ω and strict convexity of the function f , we have

$$f(z) = f\left(\frac{x+y}{2}\right) < \frac{1}{2}f(x) + \frac{1}{2}f(y) = \frac{1}{2}f(x) + \frac{1}{2}f(x) = f(x) \quad (2.1.16)$$

which contradicts the previous assumption that there were two global solutions. \square

Taking a step forward, we would like to guide readers to a broader concept than the previous categorization of an optimization problem into linear, nonlinear and convex ones; *Conic optimization* is “the problem of optimizing a linear function over the intersection of an affine space and a closed convex cone” [9]. Fundamentals on Conic optimization are followed by the optimization itself.

Definition 2.1.19 (Cone) *A cone is a set K if, for all $x \in K$ and $\eta \in \mathbb{R}_{\geq 0}$, we have*

$$\eta x \in K \quad (2.1.17)$$

“Note that cones are not necessarily convex. For example, the set $\{(x_1, x_2)^T | x_1 \geq 0 \text{ or } x_2 \geq 0\}$, which encompasses three quarters of the two-dimensional plane, is a cone.” [11] It can be visualized as a funnel with a pointed end in the two-dimensional space and as an ice cream cone in the three-dimensional space. In the context of optimization, a cone normally means a *proper cone*. Its definition slightly differs from literature to literature; here we introduce the one excerpted from [12].

Definition 2.1.20 (Proper cone) *A proper cone is a cone K which satisfies all the followings:*

- *K is convex : $\varphi x_1 + (1 - \varphi)x_2 \in K$ for any $x_1, x_2 \in K$ and $\varphi \in [0,1]$;*
- *K is closed : it contains all its limit points;*
- *K is solid : meaning that it has nonempty interior;*
- *K is pointed : if $x_{\neq 0} \in K$, then $-x \notin K$.*

Definition 2.1.21 (Convex cone) A convex cone is a set $K \subseteq \mathbb{R}^n$ if for $x_1, x_2 \in K$ and $\eta_1, \eta_2 \in \mathbb{R}_{\geq 0}$, we have

$$\eta_1 x_1 + \eta_2 x_2 \in K \quad (2.1.18)$$

Definition 2.1.22 (Conic combination) A point constructed by a conic combination of points $x_1, x_2, \dots, x_n \in \mathbb{R}^n$ is of the form

$$\sum_{i=1}^n \eta_i x_i \quad (2.1.19)$$

where $\eta_1, \eta_2, \dots, \eta_n \in \mathbb{R}_{\geq 0}$.

Definition 2.1.23 (Conic hull) A conic hull of a set S is the set of all conic combinations of its elements:

$$\text{con}(S) := \{ \sum_{i=1}^n \eta_i x_i \mid x_i \in S, \eta_i \in \mathbb{R}_{\geq 0} \} \quad (2.1.20)$$

Cones are subdivided into many different types; in this section, we introduce two of them associated with our study: *quadratic (or second-order) cones and rotated quadratic cone*. The latter concept will be used in Chapter 4. For other types, please refer to [13].

Definition 2.1.24 (Quadratic cone) The n -dimensional quadratic cone is defined as

$$\mathbb{Q}^n = \{ x = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n \mid x_1 \geq \sqrt{x_2^2 + x_3^2 + \dots + x_n^2} \} \quad (2.1.21)$$

Definition 2.1.25 (Rotated quadratic cone) An n -dimensional rotated quadratic cone is defined as

$$\mathbb{Q}^n = \{ x = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n \mid 2x_1 x_2 \geq x_3^2 + \dots + x_n^2, x_1, x_2 \geq 0 \} \quad (2.1.22)$$

The terms *Lorentz cone* and *rotated Lorentz cone*, coined after Hendrik Antoon Lorentz, are interchangeably used for quadratic and rotated quadratic cone, respectively. These two quadratic cones are illustrated in Figure 2-5. As the latter's name suggests, it can be obtained by rotating the former, in other words, by multiplying the quadratic cone by an orthogonal matrix for rotation. Details on the transformation process can be found in [13].

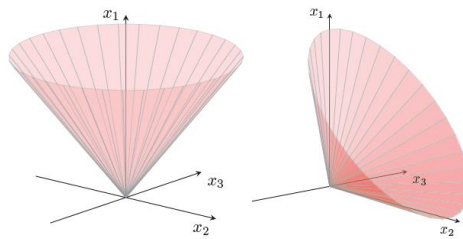


Figure 2-5 [13] Boundaries of quadratic (left) and rotated quadratic (right) cones.

Chapter II. Optimization

With all these basics, let us turn to conic optimization. It basically takes the same form as the standard form of the optimization problem (2.1.1), but with the inequality constraint $g_i(x)$ replaced by the statement which demands that a vector x should belong to a cone K .

Definition 2.1.26 (Conic optimization) *A conic optimization takes the form*

$$\begin{aligned} & \text{minimize}_x f(x) \\ & \text{subject to } x \in K \\ & \quad h_j(x) = 0, \quad j \in \mathcal{E} \end{aligned} \tag{2.1.23}$$

A conic optimization with respect to a quadratic cone is called *quadratic optimization (or second-order cone optimization)*. It can be viewed as a generalization of linear programming in that a linear objective function is optimized subject to (in)equalities with respect to variables which belong to (rotated) quadratic cone(s) [13]. This explains why some of optimization problems including the previously presented (1.4.16) and reformulated ones, which will be derived in the following chapter, are recognized as conic quadratic optimization problems even though they by definition have nothing to do with cones. This statement will become clearer in the next chapter.

2.2 Duality*

An optimization problem takes a dual form; *primal problem* and *dual problem*. The primal problem is the original problem itself, while the dual problem is defined by its *Lagrangian*. The following discussion including definitions relates to the optimization problem (2.1.1) and associated notations.

Definition 2.2.1 (Lagrangian) *The Lagrangian of the convex problem (2.1.1) $L : \mathbb{R}^n \times \mathbb{R}^p \times \mathbb{R}^q \rightarrow \mathbb{R}$ with $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $g : \mathbb{R}^p \rightarrow \mathbb{R}$ and $h : \mathbb{R}^q \rightarrow \mathbb{R}$ is*

$$L(x, \lambda, \gamma) := f(x) + \sum_{i=1}^p \lambda_i g_i(x) + \sum_{j=1}^q \gamma_j h_j(x) \tag{2.2.1}$$

As its definition indicates, a Lagrangian is a function to put objective and constraint functions together, weighting the (in)equality constraints by the *Lagrange multipliers* λ and γ , respectively. Lagrange multipliers serve as variables of the *dual function* $d(\lambda, \gamma)$, the objective function of the dual problem.

* This section is mostly based on [9], unless otherwise referenced.

Definition 2.2.2 (Dual function) The dual function $d : \mathbb{R}^p \times \mathbb{R}^q \rightarrow \mathbb{R}$ is the infimum of the Lagrangian over $D := \text{dom } f$.

$$d(\lambda, \gamma) := \inf_{x \in D} L(x, \lambda, \gamma) = \inf_{x \in D} (f(x) + \sum_{i=1}^p \lambda_i g_i(x) + \sum_{j=1}^q \gamma_j h_j(x)) \quad (2.2.2)$$

Definition 2.2.3 (Dual problem) The dual problem of the optimization problem (2.1.1) is

$$\begin{aligned} & \text{maximize}_{\lambda, \gamma} \quad d(\lambda, \gamma) \\ & \text{subject to} \quad \lambda \geq 0 \end{aligned} \quad (2.2.3)$$

Theorem 2.2.4 For any dual variables $\lambda \geq 0$ and any γ , the dual function gives a lower bound on the optimal value of the primal $f(\tilde{x})$ with a primal feasible solution \tilde{x} . That is,

$$d(\lambda, \gamma) \leq f(\tilde{x}) \quad (2.2.4)$$

proof. Let \tilde{x} belong to the feasible region of the optimization problem (2.1.1). In other words, its (in)equality constraints hold at $x = \tilde{x}$; i.e., $g_i(\tilde{x}) \leq 0$ and $h_j(\tilde{x}) = 0$ for $i \in \mathcal{E}$ and $j \in \mathcal{I}$. Then we can construct the following inequality:

$$\sum_{i=1}^p \lambda_i g_i(\tilde{x}) + \sum_{j=1}^q \gamma_j h_j(\tilde{x}) \leq 0 \quad (2.2.5)$$

Adding the objective function of the primal $f(x)$ to both sides of the inequality (2.2.5), we get

$$L(\tilde{x}, \lambda, \gamma) := f(\tilde{x}) + \sum_{i=1}^p \lambda_i g_i(\tilde{x}) + \sum_{j=1}^q \gamma_j h_j(\tilde{x}) \leq f(\tilde{x}) \quad (2.2.6)$$

Thus, by the definition of dual function, we arrive at

$$d(\lambda, \gamma) := \inf_{x \in D} L(x, \lambda, \gamma) \leq L(\tilde{x}, \lambda, \gamma) \leq f(\tilde{x}) \quad (2.2.7)$$

□

Theorem 2.2.4 shows a remarkable feature of duality; by solving the dual problem, one can estimate the optimal value of the primal one. For this reason, the dual problem is often considered as an alternative way to tackle the optimization problem especially when the primal is hard to solve. Note that, contrary to the primal, the dual problem aims to maximize its objective function, which corresponds to finding the highest lower bound on the optimal value of the primal. The highest value is indeed the best since it minimizes uncertainty. In fact, in some special cases, one can attain an exact optimal value by alternatively solving its dual. Before presenting associated concepts, let us employ the notations λ^* and γ^* to denote dual optimal solutions, just we have adopted x^* to refer to a primal optimal solution.

Definition 2.2.5 (Weak duality) We speak of weak duality when

$$d(\lambda^*, \gamma^*) \leq f(x^*) \quad (2.2.8)$$

Definition 2.2.6 (Strong duality) We speak of strong duality when

$$d(\lambda^*, \gamma^*) = f(x^*) \quad (2.2.9)$$

Definition 2.2.7 (Duality gap) The (optimal) duality gap g^* is the positive difference between $f(x^*)$ and $d(\lambda^*, \gamma^*)$, namely,

$$g^* := f(x^*) - d(\lambda^*, \gamma^*) \quad (2.2.10)$$

Based on their definitions, there is strong duality iff the duality gap is zero. Not surprisingly, this concept of strong duality has been keenly investigated and as a result there are many sufficient conditions to guarantee strong duality. Here we present one of the most widely used ones, named *Slater's condition*:

Theorem 2.2.8 (Slater's condition) For any convex problem, strong duality holds if there exists a point $x \in \text{ri}(S)$ such that

$$\begin{aligned} g_i(x) &< 0, \quad i \in I \\ h_j(x) &= 0, \quad j \in \mathcal{E} \end{aligned} \quad (2.2.11)$$

where $\text{ri}(S)$ stands for relative interior of a set S .

Proof. See Section 5.3.2 in [9].

Definition 2.2.9 [14] (Relative interior) The Relative interior of a set S is its interior relative to $\text{aff}(S)$. Any point in the set $x \in S$ belongs to $\text{ri}(S)$ if there exists a ball of radius $r \in \mathbb{R}_{>0}$ centered on x , denoted as $\mathcal{B}_r(x)$, such that

$$\mathcal{B}_r(x) \cap \text{aff}(S) \subset S \quad (2.2.12)$$

In addition to dual function and optimization, the so-called *Karush-Kuhn-Tucker* (KKT) conditions are another key components of duality. These amazingly turn the optimization problem (2.1.1) into an equivalent set of (in)equalities, in other words, into a feasibility problem.

Definition 2.2.10 (KKT conditions) Assume that the functions f, g_i and h_j in the optimization problem (2.1.1) are differentiable and strong duality (2.2.9) holds. Then the KKT conditions consist of the following (in)equalities:

$$g_i(x^*) \leq 0, \quad i \in I \quad (2.2.13)$$

$$h_j(x^*) = 0, \quad j \in \mathcal{E} \quad (2.2.14)$$

$$\lambda_i^* \geq 0, \quad i \in I \quad (2.2.15)$$

$$\lambda_i^* g_i(x^*) = 0, \quad i \in I \quad (2.2.16)$$

$$\nabla L(x^*, \lambda^*, \gamma^*) := \nabla f(x^*) + \sum_{i=1}^p \lambda_i^* \nabla g_i(x^*) + \sum_{j=1}^q \gamma_j^* \nabla h_j(x^*) = 0 \quad (2.2.17)$$

Note that the first two and the third lines are restatements of constraints of the primal and dual optimization, respectively. The fourth is termed *complementary slackness* (or *complementarity condition*) whose derivation will be given in the very next paragraph. The last one demands that the gradient of Lagrangian vanish at x^* , which reflects the fact that the optimal solution x^* minimizes the Lagrangian $L(x, \lambda^*, \gamma^*)$ over $\mathbf{dom} f$.

This is to how arrive at complementary slackness: suppose that strong duality holds, then

$$\begin{aligned} d(\lambda^*, \gamma^*) &= \inf_x \left(f(x) + \sum_{i=1}^p \lambda_i^* g_i(x) + \sum_{j=1}^q \gamma_j^* h_j(x) \right) \\ &= f(x^*) + \sum_{i=1}^p \lambda_i^* g_i(x^*) + \sum_{j=1}^q \gamma_j^* h_j(x^*) \end{aligned} \quad (2.2.18)$$

The second line must equate to $f(x^*)$ by strong duality. Given the equality constraint (2.2.14), we get

$$\sum_{i=1}^p \lambda_i^* g_i(x^*) = 0 \quad (2.2.19)$$

By additional constraints $\lambda \geq 0$ and $g_i(x) \leq 0$ with $i \in I$, every term in the summation (2.2.19) must be zero, that is, for all i ,

$$\lambda_i^* g_i(x^*) = 0 \quad (2.2.20)$$

This condition (2.2.20) is called *complementary slackness*. To summarize, for any optimization problem with differentiable objective and constraints functions, when strong duality holds, the associated optimal solutions satisfy its KKT conditions. In particular, a relation between KKT conditions and convex optimization is explained by the following theorem and corollary:

Theorem 2.2.11 *For convex problems with differentiable object and constraint functions, KKT conditions are sufficient for optimality with strong duality.*

Proof. Assume that \tilde{x} and $(\tilde{\lambda}, \tilde{\gamma})$ satisfy the KKT conditions. Then \tilde{x} is primal feasible based on the first two conditions (2.2.13) and (2.2.14). Moreover, since $\tilde{\lambda}_i \geq 0$ according to the third one (2.2.15), $L(x, \tilde{\lambda}, \tilde{\gamma})$ is convex in x . The last condition (2.2.17) states that the gradient of $L(x, \tilde{\lambda}, \tilde{\gamma})$ with respect to x vanishes at $x = \tilde{x}$, which leads to $\inf_{x \in D} L(x, \tilde{\lambda}, \tilde{\gamma}) = L(\tilde{x}, \tilde{\lambda}, \tilde{\gamma})$. From these considerations, we have

$$\begin{aligned} d(\tilde{\lambda}, \tilde{\gamma}) &:= \inf_{x \in D} L(x, \tilde{\lambda}, \tilde{\gamma}) \\ &= L(\tilde{x}, \tilde{\lambda}, \tilde{\gamma}) \\ &= f(\tilde{x}) + \sum_{i=1}^p \lambda_i g_i(\tilde{x}) + \sum_{j=1}^q \gamma_j h_j(\tilde{x}) \\ &= f(\tilde{x}) \end{aligned} \tag{2.2.21}$$

The conditions (2.2.14) and (2.2.16) are used in the last line. This correspond to the zero duality gap and therefore \tilde{x} and $(\tilde{\lambda}, \tilde{\gamma})$ are primal and dual optimal solutions, respectively. \square

Corollary 2.2.12 *For convex problems with differentiable object and constraint functions, when Slater's condition is satisfied, that is, when strong duality holds, KKT conditions are necessary and sufficient for optimality.*

Lastly, Just as a function and an optimization problem, a proper cone K (Definition 2.1.20) has its dual, the *dual cone*. Furthermore, a cone which coincides with its dual is called *self-dual*.

Definition 2.2.13 (Dual cone) *The dual cone K^* of a cone K is defined as*

$$K^* := \{ y \in \mathbb{R}^n \mid \langle y, x \rangle \geq 0 : x \in K \} \tag{2.2.22}$$

where $\langle y, x \rangle$ denotes the Euclidean inner product.

Definition 2.2.14 (Self-dual) *A cone K having its dual K^* is said to be self-dual if*

$$K = K^* \tag{2.2.23}$$

Geometrically, the dual cone can be interpreted as a set of vectors which form an acute to the right angles with any vector in the original cone. In addition to the simple one $K \in \mathbb{R}_{\geq 0}^n$, quadratic and rotated quadratic Lorentz cones, mentioned in the last section, are also self-dual. Refer to [13] for its proof.

2.3 CVX and Numerical solvers

To numerically solve optimization problems, we use *CVX*, a *Matlab*-based package for specifying and solving convex programs [15], [16], [17]. Its developers have named the mechanism behind the software ‘*disciplined convex programming*’ and give a brief description of it on their website [16]: it is based on the so-called *DCP ruleset* which identifies a type of a given optimization problem and converts it to a solvable form. This implies that an optimization problem can be treated as a different type due to the built-in ruleset; which in fact happens to our optimization problems. For detailed information on the DCP ruleset, please refer to *CVX Users’ Guide* [18]. This document can be of help when troubleshooting as well.

According to [18], CVX is highly versatile in that it is applicable to diverse types of optimization: *linear programming* (LP), *quadratic programming* (QP), *second-order cone programming* (SOCP), *semidefinite programming* (SDP), *geometric programming* (GP) and *integer programming* (IP). Moreover, it says it can solve much more complex convex optimization problems including a nondifferentiable function such as the *l1 norm*. Recall that we, at the end of the last chapter, arrive at an optimization problem whose objective function is the *l1 norm* of an HS function. Thus CVX is a reasonable choice for this study.

Additionally, CVX supports a variety of solvers; solvers named *SeDuMi* (Self-Dual Minimization) [19] and *SDPT3* [20] are built in the standard CVX distribution [16], whereas others need installing. Solvers can be easily switched through the use of the command ‘*cvx_solver solvername*’. The capabilities of some solvers that can be paired with CVX are given in Table 2-1. It shows that one should choose a solver considering the type of optimization problem.

Solver name	LP	QP	SOCP	SDP	GP	IP
SeDuMi	Y	Y	Y	Y	E	N
SDPT3	Y	Y	Y	Y	E	N
Gurobi	Y	Y	Y	N	N	Y
Mosek	Y	Y	Y	Y	Y	Y
GLPK	Y	N	N	N	N	Y

Table 2-1 [18] Different capabilities of some solvers CVX supports. Y stands for Yes, N for No, and E for Experimental.

Chapter II. Optimization

For this study, we use *Mosek* (Mathematical Optimization Software package) [21] and *SeDuMi*, which are comparable when it comes to solving from LP to SDP according to Table 2-1. Although their capabilities are comparable and they employ the same method, the so-called *primal-dual interior point method* [22], to deal with optimization problems, it turns out that they always yield slightly or noticeably different results. In the next chapter, we will present results mostly obtained using Mosek and sometimes the ones derived from SeDuMi alongside in order to highlight these discrepancies.

Regardless of solvers, one will end up with one of the status messages on the list below [18]. The ones we encountered while solving optimization problems are marked in bold. These status messages and descriptions should be understood in the context of primal-dual interior point methods, which is the subject of the next section.

- **Solved** : A complementary (primal and dual) solution has been found;
- **Unbounded** : The problem has been proven to be unbounded along the primal direction;
- Infeasible : The problem has been proven to be infeasible;
- **Inaccurate/Solved** : The problem has been solved by relaxing its conditions;
- Inaccurate/Unbounded : The problem is likely to be unbounded;
- Inaccurate/Infeasible : The problem is likely to be infeasible;
- **Failed** : The solver has failed to solve the problem.

Chapter III

Hilbert-Schmidt extrapolation function

3.1 Optimization problem

Let us now turn our attention back to the optimization problem introduced in Section 1.4.

$$\begin{aligned} \text{minimize}_f |f|_1 &:= \int_0^T |f(t)| dt \\ \text{subject to} & \left| \int_0^T f(t) e^{-iEt} dt - e^{-iE\tau} \right| \leq \delta, \quad \forall E \in [-E_{max}, E_{max}] \end{aligned} \quad (1.4.16)$$

For the sake of simplicity, we adopt the functional ansatz

$$f(t) = \sum_{j=1}^n c_j \delta(t - t_j) \quad (3.1.1)$$

and relax the inequality condition (1.2.5) to $|\Delta f(E)| \leq \delta$, for $E \in [-E_{max}, E_{max}]$. The optimization problem is then recast as

$$\begin{aligned} \text{minimize}_c & \sum_{j=1}^n |c_j| \\ \text{subject to} & \left| \sum_{j=1}^n c_j e^{-iEt_j} - e^{-iE\tau} \right| \leq \delta, \quad \forall E \in \mathbb{E} \end{aligned} \quad (3.1.2)$$

where, $c := (c_1, \dots, c_n)$ is the coefficient vector of the ansatz (3.1.1), $t_j := \frac{j-1}{r-1} T$ are time steps with $j = 1, \dots, n$ and $\mathbb{E} := \left\{ \left(-1 + 2 \frac{l-1}{r-1} \right) E_{max} : l = 1, \dots, r \right\}$ for the maximum observable time T , maximum energy E_{max} and resolution r .

3.2 Results

The optimization problem in a simpler form (3.1.2) is solved by Mosek [21] paired with CVX [16]. Results derived from Mosek will be occasionally compared with those from SeDuMi [19] in order to check their reliability. As mentioned in Section 2.3, these solvers always give slightly different results, even though both are based on the same method; as a result, we can trust three digits at most. Such discrepancies are assumed to arise from some underlying differences between the solvers in conjunction with the fact that the optimization problem is *ill-conditioned* [23]. A problem's being ill-conditioned means that great computer precision is required to obtain reliable solutions.

In the first step, for reasonable settings ($T = 1, \tau = 2, E_{max} = 1$ and $r = 100$), the delta interval where the problem is '**Solved**' without any computational issues is examined (see Section 2.3 for the list of status messages); Mosek, $\delta \geq 0.004$; and SeDuMi, $\delta \geq 0.05$. When it comes to Mosek, to be specific, all the coefficients of the ansatz (3.1.1) become zero for $\delta > 0.985$, while we get the messages '**inaccurate/Solved**' for $10^{-8} < \delta < 0.004$ and '**Failed**' for $\delta < 10^{-8}$. Whilst the first result implies that the extrapolation loses its meaning and becomes trivial when it comes to too large errors allowed, the others reveal limitations of tackling the optimization problem with the numerical solvers. This thesis includes results with the message '**Solved**' only.

The most striking result is that, for the feasible delta interval [0.004.0.985], the majority of the coefficients c_j are null and only few have nonzero values (Figure 3-1). In the light of the overall appearance of the coefficients, such nonzero terms will be referred to as '*sparse coefficients (or terms)*' henceforth. Figure 3-1 additionally shows that the number of sparse coefficients varies with delta; the smaller the delta, the more of them. While their sign alternates between plus and minus, their absolute values also increase as delta decreases, which can be seen by the increasing scale of the vertical axis. These indicate that the extrapolation becomes more complicated as the error model gets tightened. Despite the limitations of the numerical solvers, we can deduce from Figure 3-1 that sparse coefficients do not feature in the HS extrapolation function (3.1.1) as δ approximates zero.

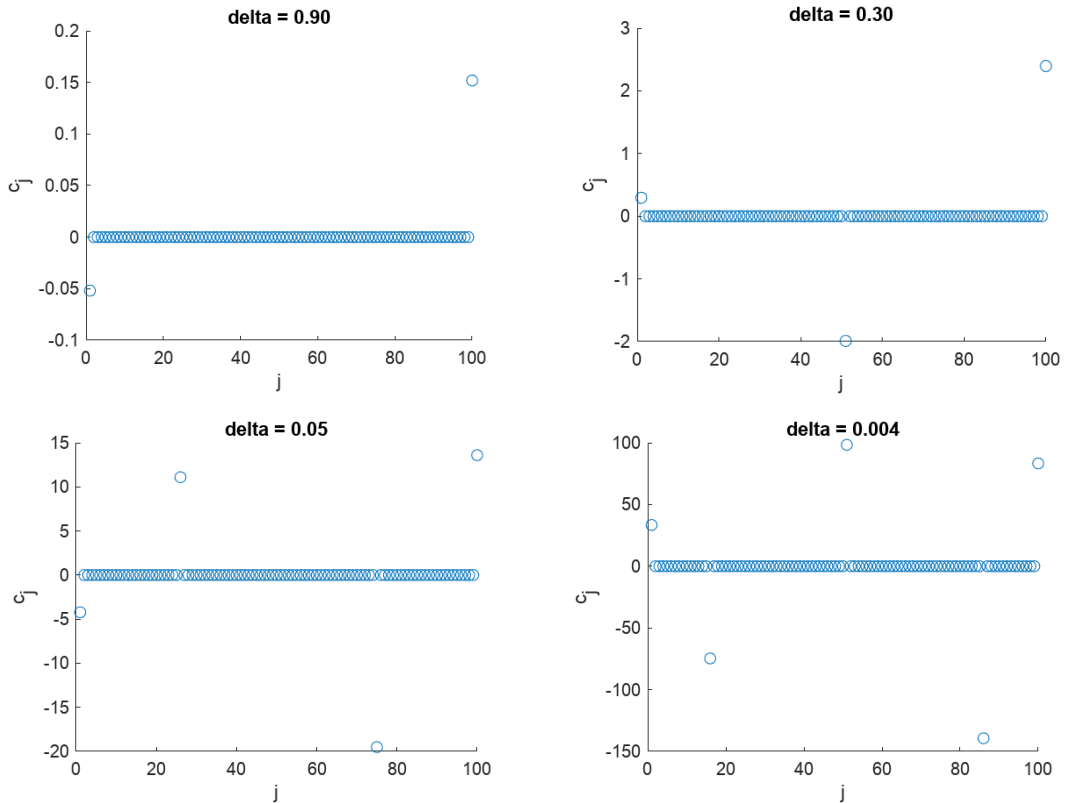


Figure 3-1 Change in an optimal solution of the problem (3.1.2) with delta. In the light of the overall appearance of the coefficients, the nonzero terms are named '*sparse coefficients*'. As delta decreases, the number of sparse coefficients increases and so do their absolute values.

For some values of delta, some neighboring terms are observed to show up together (See Figure 3-2 for an example). For the sake of simplicity, we from now on regard a local maximum or a local minimum as a sparse coefficient and assume that it takes as its value the sum of its own value and its close neighbors'. In Figure 3-2, for instance, two terms appear right next to one another (marked black): $c(30) = 2.35585$ and $c(31) = 16.31680$. Applying the approximation, we will view $c(31)$ as a sparse term whose value is $2.35585 + 16.31680 = 18.67265$. This approximation is indeed reasonable given Figure 3-3, which illustrates how CVX processes neighboring nonzero components when the resolution ('res') is set lower. Furthermore, Table 3-1 contains indices and values of sparse terms obtained using the two different solvers at $\delta = 0.004$, associated with the first two plots in Figure 3-3. Whereas their indices coincide, at most three digits of their values turn out to be comparable.

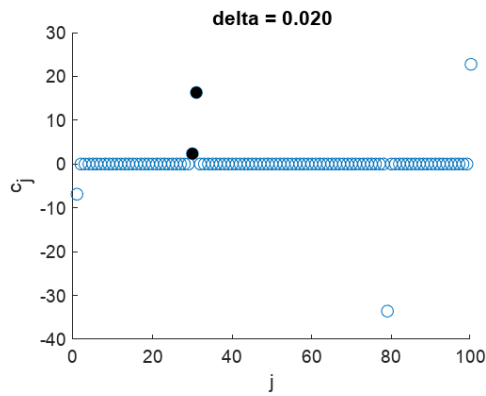


Figure 3-2 At some delta values, for example $\delta = 0.020$, neighboring non-zero terms show up (marked black).

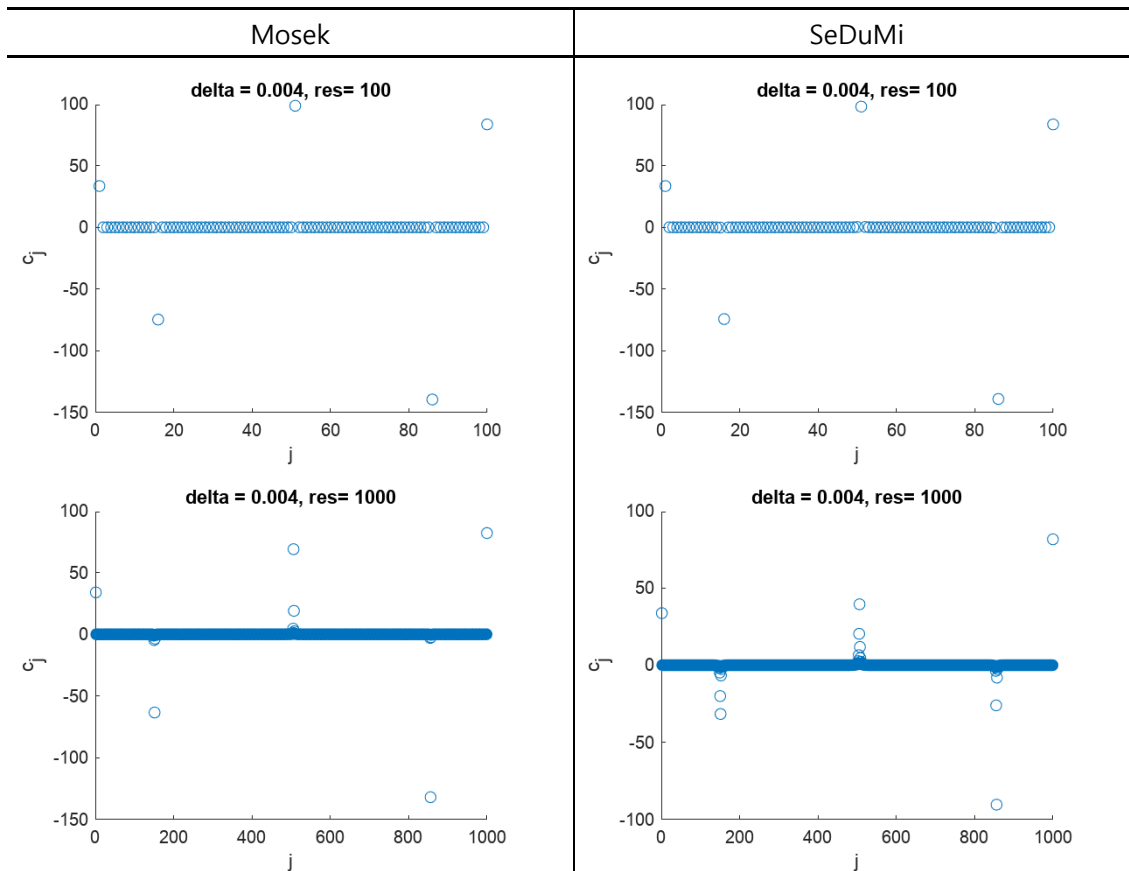


Figure 3-3 Change in optimal solutions with resolution, obtained using Mosek (*Left column*) and SeDuMi (*Right column*). This illustrates how CVX processes neighboring nonzero components when the resolution ('res') is set lower and thereby increases our trust in the approximation.

Index	Mosek	SeDuMi
1	33.41	33.38
16	-74.68	-74.28
51	98.36	97.70
86	-139.38	-138.97
100	83.29	83.26

Table 3-1 Indices and values of sparse terms associated with the first two plots taken at $\delta = 0.004$ in Figure 3-3. These are accordingly derived from Mosek and SeDuMi as well. Whereas indices of sparse terms coincide, their values are comparable up to three digits.

Based on the approximation mentioned earlier, we first study how indices of sparse terms change with delta (Figure 3-5). In order to get a better understanding of this plot, one might want to refer to Figure 3-4, which illustrates how Figure 3-5 is constructed out of plots of sparse coefficients obtained over the feasible delta interval $[0.004, 0.985]$. In Figure 3-5, we group the points into five according to their sign. Recall our previous observation of the sign of sparse terms; it alternates between plus and minus in the downhill direction of index. To be specific, while c_{100} , $m2$ and $m4$ are positive, the others are negative. The group name ' c_{100} ' reflects that the last term always exists regardless of delta. On the other hand, the others are named ' m^* ' on the ground that they tend to 'move' towards higher indices as delta decreases; they are numbered from one to four according to their order of appearance in the descent direction of delta. In the same plot, it is also observed that every m^* makes few 'pauses' in between; their indices remain unchanged within some intervals of delta. Despite lack of data over smaller values of delta, $m4$ is assumed to behave the same way. Furthermore, m^* groups are related through such 'metastable intervals' in that whenever $m1$ enters such an interval, the others start to make their appearance one by one. These phenomena can be found at three values of delta in Figure 3-5. First, at $\delta = 0.335$, $m1$ enters a metastable interval, while $m2$ appears first. Second, at $\delta = 0.014$, $m1$ and $m2$ simultaneously go in such an interval, whereas $m3$ makes its first appearance. Lastly, at $\delta = 0.081$, $m4$ shows up first when the others encounter the other interval.

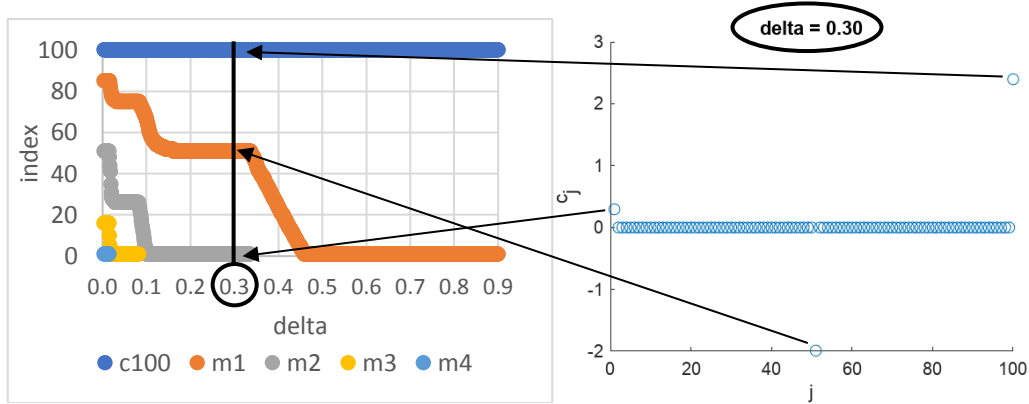


Figure 3-4 Illustration of how the index-delta plot (Figure 3-5) is constructed out of sparse coefficients obtained over the feasible delta interval.

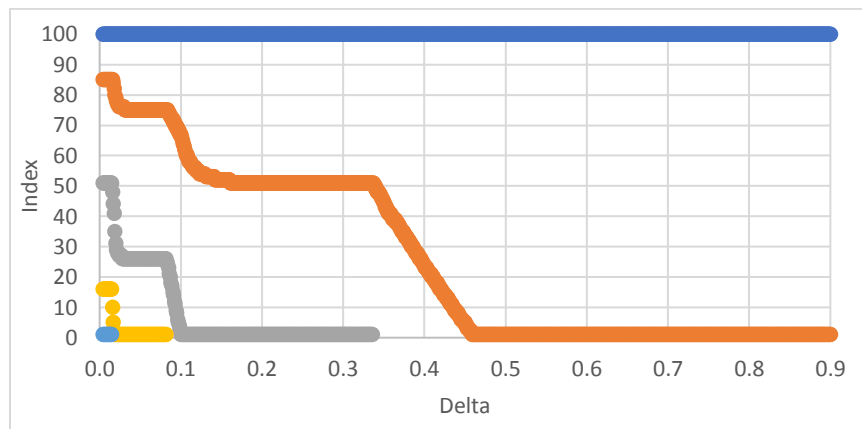


Figure 3-5 Plot of indices of sparse coefficients against delta. We group the points into five; c_{100} , $m1$, $m2$, $m3$ and $m4$. Whereas the last term always exists regardless of delta, the others tend to 'move' towards higher indices in the downhill direction of delta. m^* groups are related through 'metastable intervals' in that whenever $m1$ enters such an interval, the others start to make their appearance one by one.

Afterwards, for the ℓ_1 norm of the coefficient vector and m^* groups, their values against delta are respectively fitted using MATLAB with 95% confidence bounds on coefficients of their fit functions (Figure 3-6). Detailed formulars of the fit functions including their R-square values are given in Table 3-2. Remarkably, for $\delta < 0.020$, all the value against delta plots are fitted to linear functions (Figure 3-7 and Table 3-3), which significantly simplifies estimation of their values. All numbers in the two tables are rounded to four significant digits.

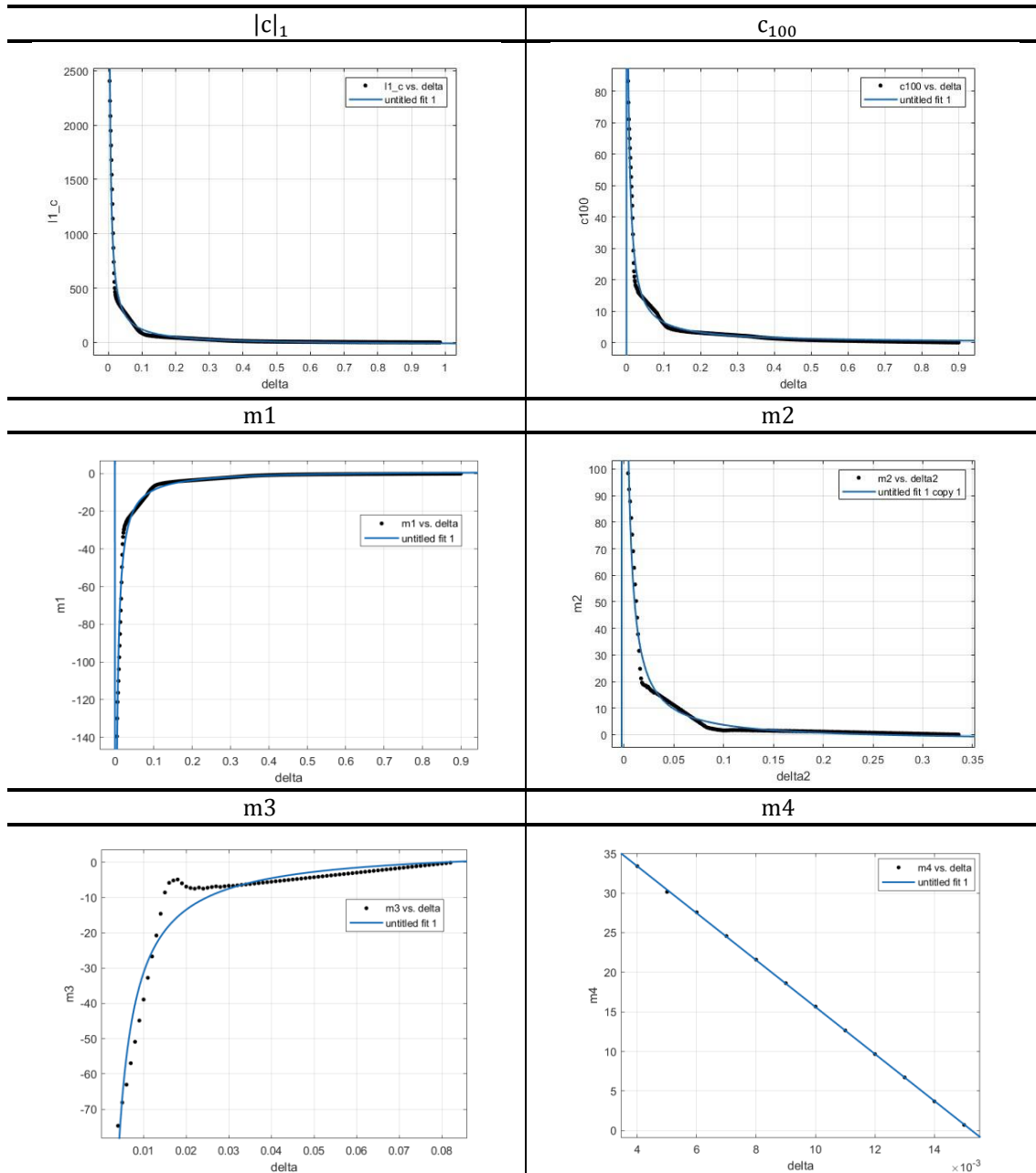


Figure 3-6 For the ℓ_1 norm of the coefficient vector and \mathbf{m}^* groups, their values against δ are respectively fitted using MATLAB with 95% confidence bounds on coefficients of their fit functions. Details of the fit functions are given in Table 3-2.

Chapter III. Hilbert–Schmidt extrapolation function

$ c _1$	c_{100}
$f(\delta) = a \cdot \delta^b + c$ $a = 17.07 (15.83, 18.31)$ $b = -0.9309 (-0.9456, -0.9161)$ $c = -25.49 (-29.00, -21.97)$ R-square: 0.9700	$f(\delta) = 1/(a \cdot \delta + b)$ $a = 1.475 (1.448, 1.502)$ $b = 0.004992 (0.004744, 0.005241)$ R-square: 0.9777
$m1$	$m2$
$f(\delta) = -1/(a \cdot \delta + b) + c$ $a = 0.9434 (0.9221, 0.9646)$ $b = 0.002542 (0.002382, 0.002703)$ $c = 1.653 (1.495, 1.812)$ R-square: 0.9782	$f(\delta) = 1/(a \cdot \delta + b) + c$ $a = 1.602 (1.527, 1.677)$ $b = 0.002334 (0.001866, 0.002802)$ $c = -2.470 (-2.827, -2.114)$ R-square: 0.9703
$m3$	$m4$
$f(\delta) = -1/(a \cdot \delta + b) + c$ $a = 2.791 (2.268, 3.314)$ $b = 7.258e-05 (-0.002397, 0.002542)$ $c = 4.462 (2.519, 6.405)$ R-square: 0.9304	$f(\delta) = a \cdot \delta + b$ $a = -2966 (-2985, -2946)$ $b = 45.26 (45.06, 45.46)$ R-square: 0.9999

Table 3-2 Details of fit functions in Figure 3-6 including their R-square values. All the numbers are rounded to four significant digits.

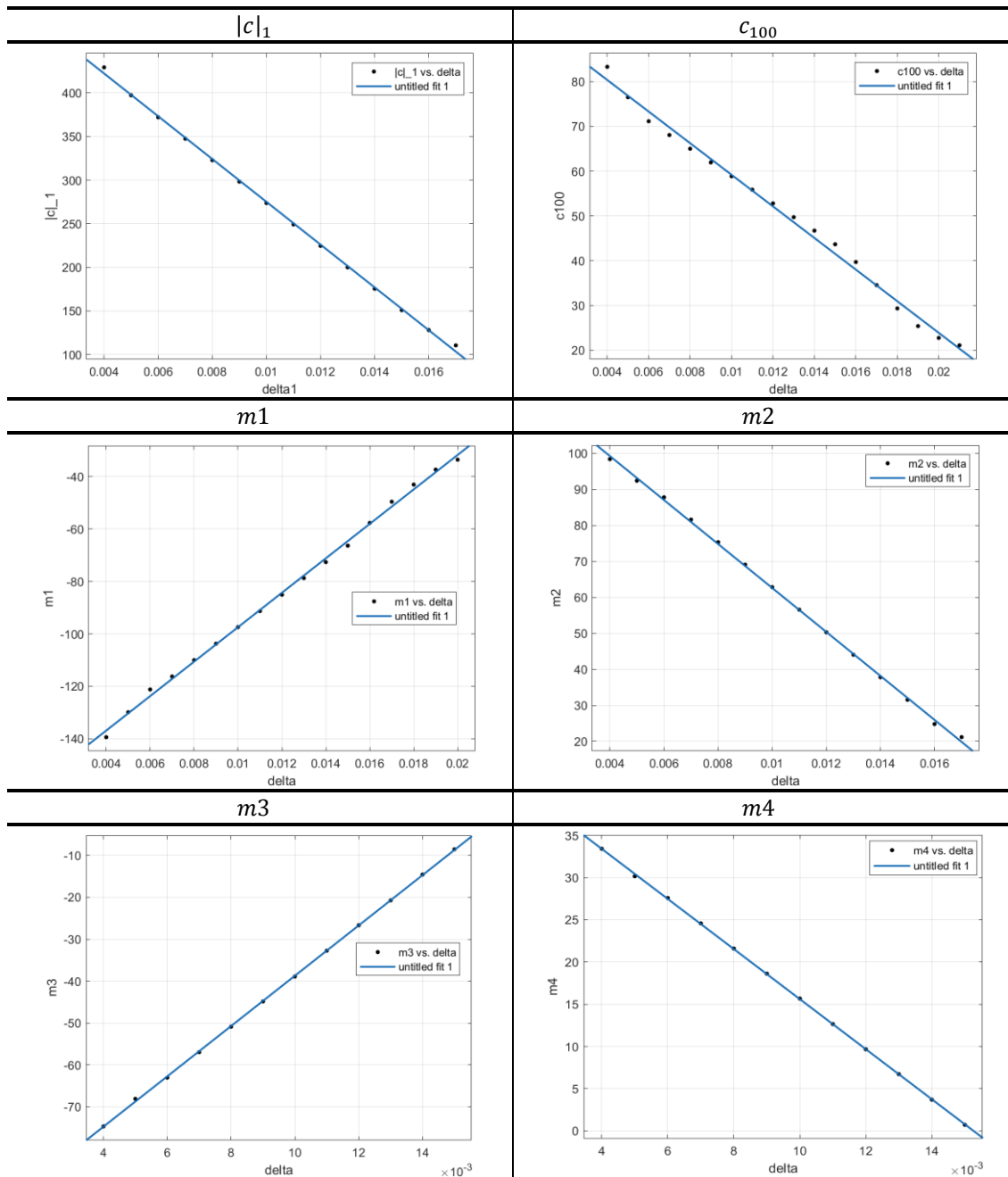


Figure 3-7 For the ℓ_1 norm of the coefficient vector and \mathbf{m}^* groups, their values against delta less than 0.020 are respectively fitted using MATLAB with 95% confidence bounds on coefficients of their fit functions. Details of the fit functions are given in Table 3-3. Note the linearity of all the fit functions.

$ c _1$	c_{100}
$f(\delta) = a^* \delta + b$ $a = -2.453e+04$ (-2.499e+04, -2.407e+04) $b = 520.4$ (515.2, 525.5) R-square: 0.9991	$f(\delta) = a^* \delta + b$ $a = -3529$ (-3680, -3379) $b = 94.48$ (92.44, 96.51) R-square: 0.9936
$m1$	$m2$
$f(\delta) = a^* \delta + b$ $a = 6569$ (6414, 6724) $b = -163.2$ (-165.2, -161.1) R-square: 0.9980	$f(\delta) = a^* \delta + b$ $a = -6107$ (-6212, -6002) $b = 123.7$ (122.5, 124.8) R-square: 0.9993
$m3$	$m4$
$f(\delta) = a^* \delta + b$ $p1 = 6003$ (5955, 6052) $p2 = -98.76$ (-99.25, -98.28) R-square: 0.9999	$f(\delta) = a^* \delta + b$ $a = -2966$ (-2985, -2946) $b = 45.26$ (45.06, 45.46) R-square: 0.9999

Table 3-3 Details of fit functions in Figure 3-7 including their R-square values. All the numbers are rounded to four significant digits.

By taking the same steps with respect to tau, we can extend the ansatz (3.1.1) to tau, i.e., $f_\delta \rightarrow f_{\delta,\tau}$. During the first phase, we find reasonable settings ($T = 1, \delta = 0.004, E_{max} = 1$ and $r = 100$) and check the feasible tau interval: $\tau \in [1.0, 2.5]$. Figure 3-8 shows how indices and values of sparse coefficients change with tau. The number of sparse coefficients and their absolute values significantly increase with tau, which suggests that the extrapolation function becomes more complicated at a future time point farther away from the observable time interval. On the other hand, the signs of sparse terms follow the same rule as before; their sign alternates between plus and minus in the descent direction of index. Note that, at $\tau = 2.5$, there exist six sparse coefficients in total according to the approximation introduced before. As before, Figure 3-9, the index versus tau plot over the feasible tau interval is obtained by aggregating plots as those in Figure 3-8. Above all, the most significant result to emerge from tau-related data is the appearance of a new group named ‘**m5**’, which supports our previous deduction that the ansatz ends up with many nonzero coefficients as δ approximates zero. Taking the existence of **m5** into consideration, the index-delta plot is drawn again with respect to deltas close to zero in Figure 3-10. Completing Figure 3-5, this plot illustrates how indices of sparse coefficients change within the interval of delta where the optimization problem is ‘**Solved**’ by the numerical solver.

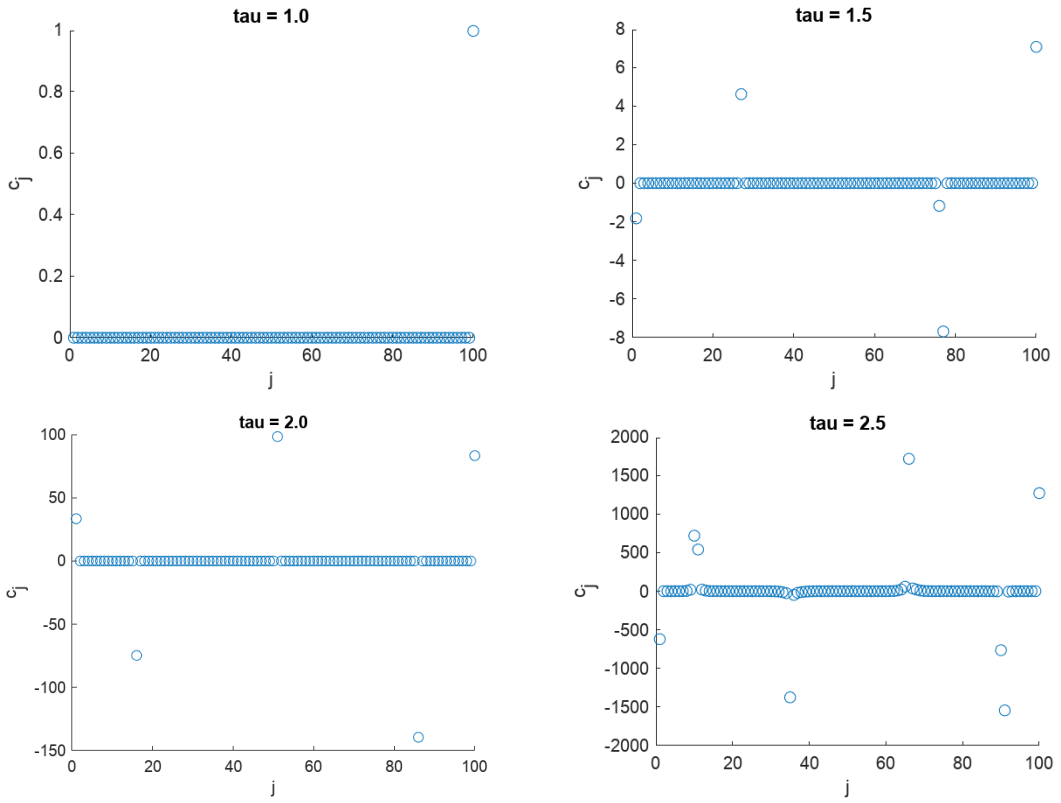


Figure 3-8 Change in an optimal solution of the problem (3.1.2) with τ . The so-called sparse coefficients are observed as well. Their number and absolute values increase with τ , while the sign of sparse coefficients alternates between plus and minus. Note that, at $\tau = 2.5$, there exist six sparse coefficients in total according to the approximation introduced before.

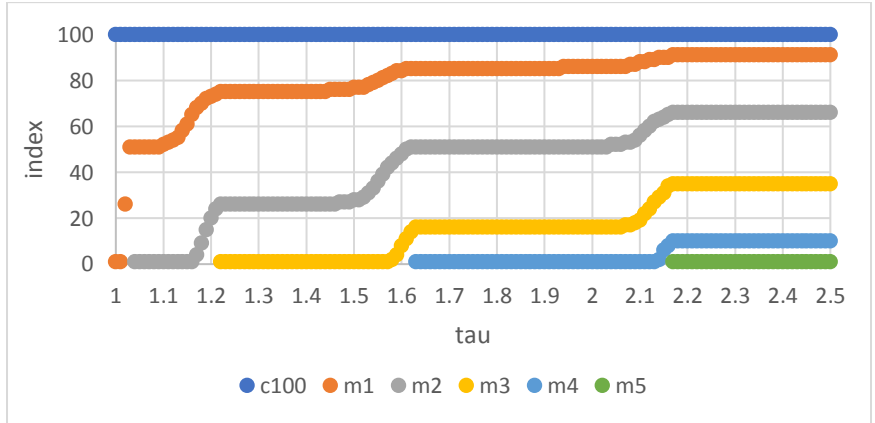


Figure 3-9 Plot of indices of sparse coefficients against τ over the feasible τ interval. Note the appearance of a new index group, **m5**.

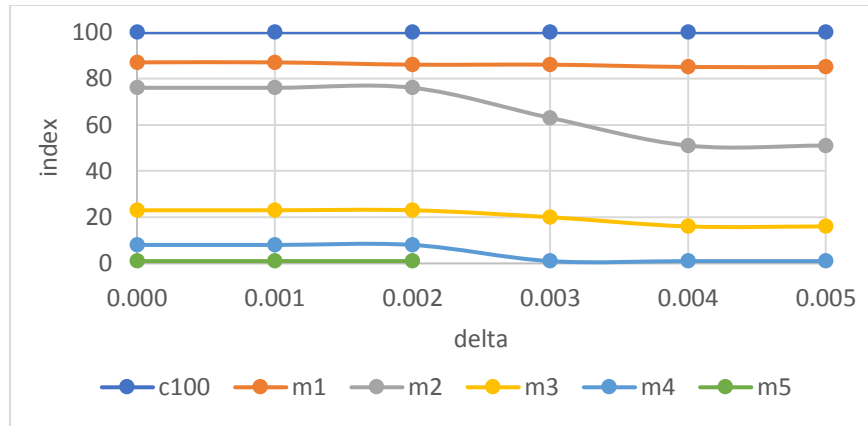


Figure 3-10 The index-delta plot over small values of delta, which completes Figure 3-5. Though the numerical solver expects up to six sparse coefficients, as delta approximates zero, more and more nonzero terms are assumed to show up to the point the ansatz (3.1.1) rather looks continuous.

Afterwards, with respect to each index group, the value against tau plot is fitted using MATLAB in the same way as before. Combined with those in Figure 3-6, we obtain fit functions which take both delta and tau as their arguments and thereby values of index groups — \mathbf{c}_{100} , $\mathbf{m1}$, ..., and $\mathbf{m5}$ — can be estimated to some degree of reliability (see Appendix B). We are, however, aware that this approach has two limitations. First, we can hardly predict their indices, however accurately their values are predicted by the fit functions. Second, our knowledge is just limited to the six index groups found by Mosek. Though we have deduced that the majority of coefficients of the ansatz (3.1.1) have nonzero values as delta approximates zero, there is actually no way to estimate indices and values of index groups other than the six.

Chapter IV

Error models

4.1 Optimization problem

Removing the assumption that the error model is constant, the optimization problem (1.4.16) is written as

$$\begin{aligned} & \text{minimize}_{f,\varepsilon} \int_0^T |f(t)|\varepsilon(t) dt \\ & \text{subject to } |\Delta(E)| \leq \delta, \quad \forall E \in [-E_{max}, E_{max}] \end{aligned} \quad (4.1.1)$$

Let us recast this problem with new variables $\Omega \in \mathbb{R}_{>0}$ and $\nu := \delta^2$ as follows:

$$\begin{aligned} & \text{minimize}_{f,\varepsilon,\nu} \int_0^T |f(t)|\varepsilon(t) dt + \Omega\nu \\ & \text{subject to } |\Delta(E)|^2 \leq \nu, \quad \forall E \in [-E_{max}, E_{max}] \end{aligned} \quad (4.1.2)$$

Given an extrapolation function $f(t)$, this problem (4.1.2) can be used to find a corresponding error model $\varepsilon(t)$ and Ω . To this end, going through some steps as given in Appendix C, we arrive at the optimization problem:

$$\begin{aligned} & \text{minimize}_{f,\mu,\varepsilon,\nu} \int_0^T \mu(t)\varepsilon(t) dt + \Omega\nu \\ & \text{subject to} \\ & \mu(t) - f(t) \geq 0, \mu(t) + f(t) \geq 0, \quad \forall t \in [0, T] \\ & \left(\nu, \frac{1}{2}, \int f(t) \cos(Et) dt - \cos(E\tau), \int f(t) \sin(Et) dt - \sin(E\tau) \right) \in K_{sqrt}, \quad \forall E \in [-E_{max}, E_{max}] \end{aligned} \quad (4.1.3)$$

where K_{sqrt} denotes the rotated quadratic cone (Definition 2.1.25). Since the rotated quadratic cone is self-dual (Definition 2.2.14), the dual of the problem (4.1.3) reads

$$\begin{aligned} & \text{maximize}_{\nu^+, \nu^-, Z, \varepsilon} \int \langle Z(E), \left(0, \frac{1}{2}, \cos(\tau E), \sin(\tau E)\right) \rangle dE \\ & \text{subject to} \\ & \varepsilon(t) - V^+(t) - V^-(t) = 0 \\ & V^+(t) - V^-(t) - \int \langle Z(E), (0, 0, \cos(Et), \sin(Et)) \rangle dE = 0 \\ & \Omega - \int \langle Z(E), (1, 0, 0, 0) \rangle dE = 0 \\ & Z(E) \in K_{sqrt}, \quad \forall E \in [-E_{max}, E_{max}] \\ & V^+(t), V^-(t) \geq 0, \quad \forall t \in [0, T] \end{aligned} \quad (4.1.4)$$

and the duality gap is given by

$$\begin{aligned} \gamma(V^+, V^-, Z, f, \mu, \nu) &:= \int (\mu(t) - f(t) V^+(t)) dt + \int (\mu(t) + f(t) V^-(t)) dt \\ &+ \int \langle Z(E), \left(\nu, \frac{1}{2}, \int f(t) \cos(Et) dt - \cos(E\tau), \int f(t) \sin(Et) dt - \sin(E\tau) \right) \rangle dE \end{aligned} \quad (4.1.5)$$

Let us define

$$\bar{\Delta} := \max_{E \in [-E_{max}, E_{max}]} \Delta_f(E) \quad (4.1.6)$$

If one finds feasible $V^+, V^-, Z, \varepsilon(t)$ and Ω such that

$$\gamma(V^+, V^-, Z, f, |f(t)|, \bar{\Delta}^2) = 0 \quad (4.1.7)$$

this means that $f(t)$ is an optimal HS extrapolation function for $\varepsilon(t)$ and Ω . In this case, $f(t)$ is also an optimal extrapolation function for $\{\mu\varepsilon(t_j) \geq 0\}$ and $\mu\Omega$ for all $\mu \in \mathbb{R}_{>0}$. Thus it is reasonable to set $\Omega = 1$. These considerations bring us to the optimization problem:

$$\begin{aligned} & \text{minimize}_{V^+, V^-, Z, \varepsilon} \gamma(V^+, V^-, Z, f, |f(t)|, \bar{\Delta}^2) \\ & \text{subject to} \\ & \varepsilon(t) - V^+(t) - V^-(t) = 0 \\ & V^+(t) - V^-(t) - \int \langle Z(E), (0, 0, \cos(Et), \sin(Et)) \rangle dE = 0 \\ & 1 - \int \langle Z(E), (1, 0, 0, 0) \rangle dE = 0 \\ & Z(E) \in K_{sqr}, \quad \forall E \in [-E_{max}, E_{max}] \\ & \varepsilon(t) \geq 0, \quad \forall t \in [0, T] \\ & V^+(t), V^-(t) \geq 0, \quad \forall t \in [0, T] \end{aligned} \quad (4.1.8)$$

Considering the functional ansatz (3.1.1) and approximating the integrals over energy in the constraints in the problem (4.1.8) to the sum over the elements of $\mathbb{E} := \left\{ \left(-1 + 2 \frac{l-1}{r-1} \right) E_{max} : l = 1, \dots, r \right\}$, we have

$$\begin{aligned} & \text{minimize}_{V^+, V^-, Z, \varepsilon} \gamma_\varepsilon(V^+, V^-, Z, f) \\ & \text{subject to} \\ & \varepsilon_j - V_j^+ - V_j^- = 0 \\ & V_j^+ - V_j^- - \sum_{E \in \mathbb{E}} \langle Z(E), (0, 0, \cos(Et), \sin(Et)) \rangle = 0 \\ & 1 - \sum_{E \in \mathbb{E}} \langle Z(E), (1, 0, 0, 0) \rangle = 0 \\ & Z(E) \in K_{sqr}, \quad \forall E \in \mathbb{E} \\ & \varepsilon_j \geq 0, \quad \forall j \\ & V_j^+, V_j^- \geq 0, \quad \forall j \end{aligned} \quad (4.1.9)$$

with

$$\begin{aligned} \gamma_\varepsilon(V^+, V^-, Z, f) &:= \sum_j (|c_j| - c_j) V_j^+ + \sum_j (|c_j| - c_j) V_j^- \\ &+ \sum_{E \in \mathbb{E}} \langle Z(E), (\bar{\Delta}^2, \frac{1}{2}, \sum_j c_j \cos(Et_j) - \cos(E\tau), \sum_j c_j \sin(Et_j) - \sin(E\tau)) \rangle \end{aligned} \quad (4.1.10)$$

If its solution approximates zero, $f(t)$ can make an optimal HS extrapolation function with the error model $\varepsilon(t)$ for given parameters T, τ and E_{max} .

4.2 Potential optimal extrapolation functions

Three functions — the *Lagrange polynomial* and the two HS extrapolation functions introduced in Section 1.3 — are considered as optimal extrapolation functions for some error models. When it comes to the Lagrange polynomial, though it is originally meant to interpolate, we extend its use to extrapolation in this study.

(a) Lagrange polynomial

Definition 4.2.1 [24] (Lagrange polynomial) *Given a set of data points (x_i, y_i) with $i = 0, \dots, n$, its Lagrange (interpolating) polynomial is defined as*

$$LP_n(x) := \sum_{i=0}^n y_i \prod_{k \neq i} \left(\frac{x - x_k}{x_i - x_k} \right) \quad (4.2.1)$$

Example Consider the quadratic function $q(x) = 3x^2 + x + 2$ which passes through the three points: $(x_0, y_0) = (0, 2)$, $(x_1, y_1) = (1, 6)$ and $(x_2, y_2) = (-1, 4)$. Then the Lagrange polynomial to interpolate these points is given by

$$\begin{aligned} LP_2(x) &= \sum_{i=0}^2 y_i \prod_{k \neq i} \left(\frac{x - x_k}{x_i - x_k} \right) \\ &= y_0 \left(\frac{x - x_1}{x_0 - x_1} \right) \left(\frac{x - x_2}{x_0 - x_2} \right) + y_1 \left(\frac{x - x_0}{x_1 - x_0} \right) \left(\frac{x - x_2}{x_1 - x_2} \right) + y_2 \left(\frac{x - x_0}{x_2 - x_0} \right) \left(\frac{x - x_1}{x_2 - x_1} \right) \\ &= 2 \left(\frac{x - 1}{0 - 1} \right) \left(\frac{x - (-1)}{0 - (-1)} \right) + 6 \left(\frac{x - 0}{1 - 0} \right) \left(\frac{x - (-1)}{1 - (-1)} \right) + 4 \left(\frac{x - 0}{-1 - 0} \right) \left(\frac{x - 1}{-1 - 1} \right) \\ &= -2(x - 1)(x + 1) + 3x(x + 1) + 2x(x - 1) \\ &= 3x^2 + x + 2 \end{aligned} \quad (4.2.2)$$

which coincides with $q(x)$.

Coming back to our discussion on extrapolating the average values (1.1.1), Lagrange polynomial tries to fit the general function $a(t)$ to a polynomial $\tilde{a}(\tau)$. That is,

$$\tilde{a}(\tau) = \sum_j a(t_j) \prod_{k \neq j} \left(\frac{\tau - t_k}{t_i - t_k} \right) \quad (4.2.3)$$

Define

$$c_j := \prod_{k \neq j} \left(\frac{\tau - t_k}{t_i - t_k} \right) \quad (4.2.4)$$

Then it follows that

$$\begin{aligned} \tilde{a}(\tau) &= \sum_j a(t_j) c_j \\ &= \int_0^T f(t) a(t) dt \end{aligned} \quad (4.2.5)$$

The ansatz (3.1.1) is used in the second line.

(b) HS extrapolation function based on superoscillations

In Section 1.3, we derive the HS extrapolation function inspired by superoscillations [3], [4].

$$f_S(t) := \sum_{j=0}^N (c_S)_j \delta(t - (t_S)_j) \quad (1.3.9)$$

with

$$\begin{aligned} (c_S)_j &:= \binom{N}{j} \left(1 - \frac{\tau}{T}\right)^{N-j} \left(\frac{\tau}{T}\right)^j \\ (t_S)_j &:= \frac{T}{N} j \end{aligned} \quad (1.3.10)$$

There we consider two approximations based on two different definitions of the exponential function to recover the superoscillating Fourier sequence (1.3.4) from $e^{-iE\tau}$ under the condition $N \gg |E|\max(T, \tau)$. For reasonable settings ($T = 1, \tau = 2$ and $E = 1$), the first one $e^{-i\tau E} \approx \left(1 - \frac{1}{N} i\tau E\right)^N$ requires N to be sufficiently large, while the second one $e^{-iE\frac{T}{N}} \approx 1 - iE\frac{T}{N}$ holds for all $N \in \mathbb{N}$. Figure 4-1 shows how the approximation error between $A := e^{-iE\tau}$ and $B := \left(1 - \frac{1}{N} i\tau E\right)^N$ changes with N .

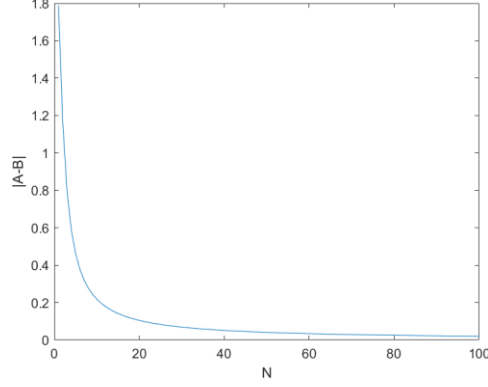


Figure 4-1 Change in the approximation error with N for the reasonable settings ($T = 1, \tau = 2$ and $E = 1$). Here A and B denote $e^{-iE\tau}$ and $\left(1 - \frac{1}{N}i\tau E\right)^N$, respectively. It indicates that N needs to be sufficiently large to make a reasonable approximation.

(c) HS extrapolation function based on McLaurin expansion

Next, recall the HS extrapolation function based on McLaurin expansion [5] and numerical differentiation [6]:

$$f_M(t) := \sum_{j=0}^N \sum_{k=0}^j \frac{\tau^j}{j! h^j} \binom{j}{k} (-1)^{j+k} \delta(t - kh) \quad (1.3.17)$$

This can be written in a similar form as the previous ones as

$$f_M(t) := \sum_{j=0}^N \sum_{k=0}^j (c_M)_{jk} \delta(t - (t_M)_k) \quad (4.2.6)$$

with

$$\begin{aligned} (c_M)_{jk} &:= \frac{\tau^j}{j! h^j} \binom{j}{k} (-1)^{j+k} \\ (t_M)_k &:= kh \end{aligned} \quad (4.2.7)$$

Since the optimization problem (4.1.9) requires the coefficient vector c_M to be a column vector, the extrapolation function (4.2.6) needs to be further reduced to

$$f_M(t) := \sum_{k=0}^j (\mathbf{C}_M)_k \delta(t - (t_M)_k) \quad (4.2.8)$$

with

$$\begin{aligned} (\mathbf{C}_M)_k &:= \sum_j (c_M)_{jk} \\ (t_M)_k &:= kh \end{aligned} \quad (4.2.9)$$

The new coefficient vector $(\mathbf{C}_M)_k$ can be understood as a vector constructed by summing up all elements in each column of the matrix $(c_M)_{jk}$.

Recall that two approximations are considered to derive the extrapolation function (1.3.17): the McLaurin expansion (1.3.14) and the numerical differentiation (1.3.16). In this section, given initial settings ($\tau = 2, E = 1$), we will check under which conditions these are reasonably valid one by one. First, Figure 4-2 illustrates a change in the approximation error between $A := e^{-iE\tau}$ and its McLaurin expansion $M := \sum_{j=0}^N \frac{d^j(e^{-iEt})}{dt^j} \Big|_{t=0} \frac{\tau^j}{j!}$ with N .

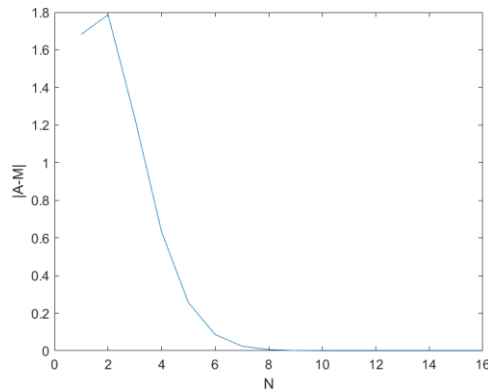


Figure 4-2 Change in the approximation error with N for the reasonable settings ($\tau = 2$ and $E = 1$). Here A and M denote $e^{-iE\tau}$ and its McLaurin expansion $\sum_{j=0}^N \frac{d^j(e^{-iEt})}{dt^j} \Big|_{t=0} \frac{\tau^j}{j!}$, respectively.

Next, we study the approximation error between the McLaurin expansion of $e^{-iE\tau}$, denoted M , and its numerical differentiation $D := \frac{1}{h^j} \sum_{k=0}^j \binom{j}{k} (-1)^{j+k} e^{-iEkh}$. Figure 4-3 illustrates how the approximation error varies with $1/h$ with respect to four different values of j . Whereas the approximation error is observed to improve with $1/h$ for small values of j as expected, some fluctuation in the error is observed for $j \geq 7$. This can be attributed to the computer precision and consequent rounding errors. For this reason, together with the fact that the parameter j is bounded by N by definition, N larger than 6 should be avoided.

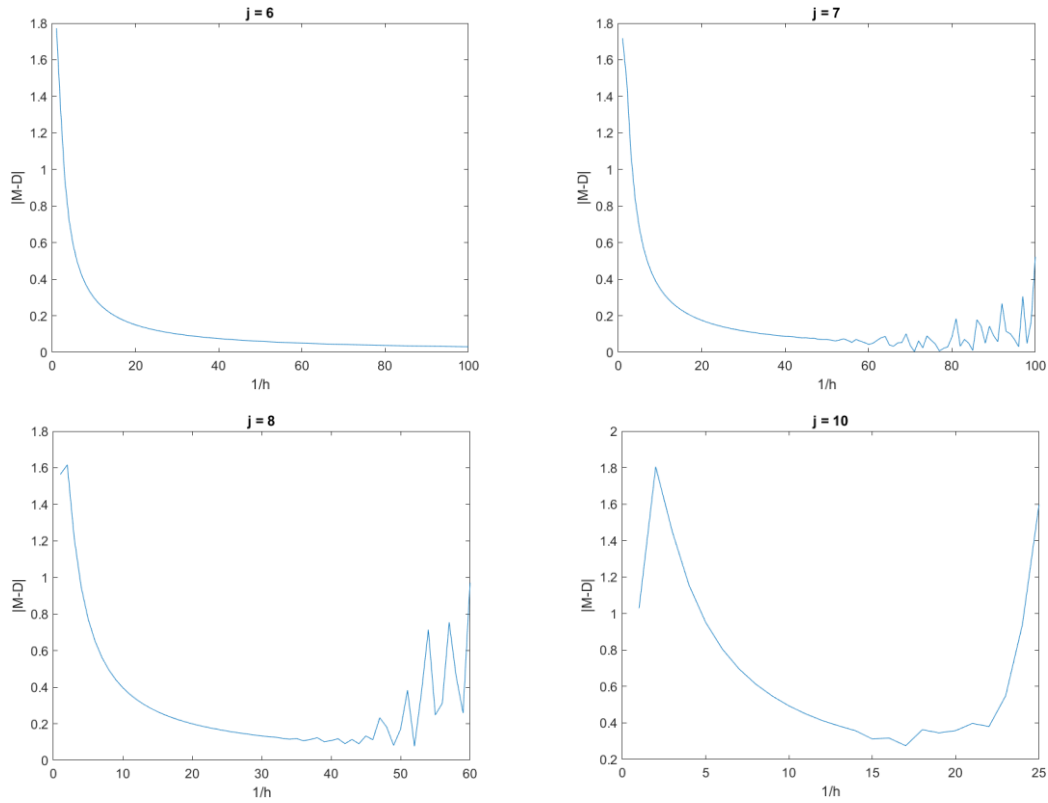


Figure 4-3 Change in the approximation error with $1/h$ and j for the reasonable settings ($\tau = 2$ and $E = 1$). Here M and D denote the McLaurin expansion $e^{-iE\tau}$ and its numerical differentiation $\sum_{j=0}^N \frac{d^j(e^{-iE\tau})}{dt^j} \Big|_{t=0} \frac{\tau^j}{j!}$, respectively.

4.3 Results

With respect to the three candidate functions presented in the last section, we solve the optimization problem (4.1.9) by using Mosek paired with CVX. Results derived from the functions share the same characteristics. To begin with, all the tables in this section indicate that both δ and γ_ε decrease with r ; the decrease in γ_ε , equivalent to improvement in the optimality of the function, is a natural consequence of dividing the given time interval more finely and thereby having more data points to refer to when fitting the Lagrange polynomial to the data points. Second, for the lower value of τ , the optimality of the functions improves on the whole as seen in Table 4-2, Table 4-5 and Table 4-8. This can be explained by the fact that extrapolation of the data at a time point closer to the observable interval $[0,1]$ is more feasible. Third, the tables concerning $E_{max} = 2$ imply that the increased uncertainty in E makes the extrapolation much harder (Table 4-3, Table 4-6, and Table 4-9).

For the values of the parameters r and $N(= r + 1)$ larger than those on the tables, we get the message '**Unbounded**' with warnings of too large numbers. These warnings concern elements of the coefficient vectors of the candidate functions, which rapidly increase with the parameters by their definitions (4.2.4), (1.3.10) and (4.2.9), respectively. Therefore, together with their own limitations in extrapolation, their use as extrapolation functions is quite restricted; they work only within a limited range of the parameters.

Next, given that the Lagrange polynomial leads to the lowest value of γ_ε among them, we can conclude that it is the most optimal. Remarkably, we observe from Table 4-1 and Table 4-2 that we can study smaller values of delta ($6.10E - 07$ at lowest) with the Lagrange polynomial than we can do with Mosek, which works for $\delta \geq 0.004$ (see Section 3.2).

Lastly, we find out that all the functions lead to optimal values close to zero and approximately correspond with the zero-error model, a column vector whose components are all zero; error models associated with the tables in this section are available in Appendix D. We did not expect to find the same optimal error model for all the tested functions. On the other hand, it is not a complete surprise that the zero-error model can fit widely different functions, as it corresponds to an optimization problem where the only goal is minimizing the extrapolation error. In this regard, further studies need to be performed. As one possible way to come up with HS extrapolation functions which match error models other than the zero-error one, we suggest tracking back the coefficient vectors corresponding to different types of error models. This idea is elaborated in Appendix E.

(a) Lagrange polynomial

r	δ	γ_ε
4	1.78E-03	3.17E-02
5	5.31E-03	2.81E-03
6	1.31E-03	1.72E-04
7	2.80E-03	7.73E-06
8	7.81E-04	6.10E-07

Table 4-1 Changes in δ and γ_ε with r for $(T, \tau, E_{max}) = (1, 2, 1)$. Both δ and γ_ε decrease with r .

r	δ	γ_ε
4	2.97E-02	8.82E-04
5	5.80E-03	3.30E-05
6	9.25E-04	2.61E-06
7	1.27E-04	8.50E-07
8	3.65E-05	7.37E-09

Table 4-2 Changes in δ and γ_ε with r for $(T, \tau, E_{max}) = (1, 1.5, 1)$. Overall improvement in the optimality is observed compared to when $\tau = 2$ (Table 4-1).

r	δ	γ_ε
4	2.53	6.41
5	1.55	2.41
6	7.83E-01	6.14E-01
7	3.35E-01	1.12E-01
8	1.25E-01	1.57E-02

Table 4-3 Changes in δ and γ_ε with r for $(T, \tau, E_{max}) = (1, 2, 2)$. Overall deterioration in the optimality is observed compared to when $E_{max} = 1$ (Table 4-1).

(b) HS extrapolation function based on superoscillations

N	δ	γ_ε
10	1.04E-01	1.09E-02
11	9.47E-02	8.96E-03
12	8.65E-02	7.48E-03
13	7.98E-02	6.37E-03
14	7.55E-02	5.70E-03

Table 4-4 Changes in δ and γ_ε with r for $(T, \tau, E_{max}) = (1, 2, 1)$. Both δ and γ_ε decrease with N .

N	δ	γ_ε
11	3.46E-02	1.19E-03
12	3.17E-02	1.00E-03
13	2.92E-02	8.54E-04
14	2.71E-02	7.35E-04
15	2.53E-02	6.39E-04

Table 4-5 Changes in δ and γ_ε with r for $(T, \tau, E_{max}) = (1, 1.5, 1)$. Overall improvement in the optimality is observed compared to when $\tau = 2$ (Table 4-4).

N	δ	γ_ε
11	4.27E-01	1.82E-01
12	3.87E-01	1.50E-01
13	3.54E-01	1.25E-01
14	3.25E-01	1.05E-01
15	9.01E-02	5.02E-02

Table 4-6 Changes in δ and γ_ε with r for $(T, \tau, E_{max}) = (1, 2, 2)$. Overall deterioration in the optimality is observed compared to when $E_{max} = 1$ (Table 4-4).

(c) HS extrapolation function based on McLaurin expansion

When it comes to this function, we first figure out appropriate values for N for each set of parameters — $(T, \tau, E_{max}) = (1,2,1)$, $(1,1.5,1)$ and $(1,2,2)$, taking the two approximation methods (1.3.14) and (1.3.16) into account: for the first two, $N = 6$; and for the last, $N = 10$. With these values of N , the best optimality, equivalent to the lowest γ_ε , is attained for each case.

$1/h$	δ	γ_ε
7	1.09E-02	1.21E-02
8	9.40E-02	8.83E-03
9	8.04E-02	6.45E-03
10	7.12E-02	5.06E-03
11	3.08E-02	9.51E-04

Table 4-7 Changes in δ and γ_ε with $1/h$ for $(N, T, \tau, E_{max}) = (6, 1, 2, 1)$. Both δ and γ_ε decrease with $1/h$.

$1/h$	δ	γ_ε
9	7.72E-02	5.95E-03
10	6.96E-02	4.83E-03
11	6.21E-02	3.86E-03
12	5.65E-02	3.19E-03
13	5.43E-02	2.94E-03

Table 4-8 Changes in δ and γ_ε with $1/h$ for $(N, T, \tau, E_{max}) = (6, 1, 1.5, 1)$. An overall worsening in the optimality is observed compared to when $\tau = 2$ (Table 4-7).

$1/h$	δ	γ_ε
3	7.30E-02	5.33E-01
4	5.61E-02	3.15E-01
5	4.62E-02	2.13E-01
6	4.00E-02	1.60E-01
7	2.89E-02	8.36E-02

Table 4-9 Changes in δ and γ_ε with $1/h$ for $(N, T, \tau, E_{max}) = (10, 1, 2, 2)$. Overall deterioration in the optimality is observed compared to when $E_{max} = 1$ (Table 4-7).

V. Conclusion

In this study, we have formulated the extrapolation of time averages of HS observables as an optimization problem. Assuming an ansatz on the extrapolation functions, we solved this problem using the software packages Mosek and CVX. For reasonable settings, its optimal solution consists of few non-zero terms, the so-called 'sparse coefficients'. For moderately low values of the approximation error δ , optimal functions have at most five non-zero terms. We observed that they present alternating signs and have studied how they vary with δ . When it comes to their indices, though we could have taken an analytics guess based on their characteristic behavior, there seemed no convincing way to do that. Nevertheless, from Figure 3-5, we could deduce that the coefficient vector would have mostly non-zero terms as δ approximates zero. As for their values, for $\delta < 0.020$, their values against δ are all fitted to linear functions (Figure 3-7 and Table 3-3). Hereby we have demonstrated that the extrapolation with high accuracy is feasible to some extent, namely, for the five index groups. This remains valid after extending the ansatz to the future time point τ as given in Appendix B. In the meantime, while taking the same steps with respect to τ , we have discovered an additional index group, which supports our previous deduction. In spite of all these findings, our study is admittedly still insufficient for a complete extrapolation; due to limitations of the software, we were not able to study the optimal solution for $\delta < 0.004$. Though not included in this thesis, we have tried the *gradient descent method* [8] and *Adam* [25] as follow-up studies. However, none of them helped us to get a better understanding of the optimal solution, failing to reach δ smaller than Mosek's limit. For this reason, further studies need to be performed to either find or devise a better approach. Once these limitations are overcome, our method could be used to estimate HS observables with high precision. In addition, to further our research, we intend to extend our consideration from HS observables to any bounded observables and find universal extrapolation functions.

Afterwards, to study the association between extrapolation functions and error models, we have recast the problem (1.4.6), removing the assumption that the error model is a constant and using the notion of duality as elaborated in Appendix C. The reformulated problem (4.1.9) takes as its objective function the duality gap, which makes an indicator of the optimality of a given function. This problem is solved by Mosek paired with CVX as before, with respect to the following three extrapolation functions explained in Section 4.2: Lagrange polynomial and HS extrapolation functions based on superoscillations and McLaurin expansion, respectively. For all the candidate functions, despite some computational limitations, we have found that their optimality improves with the parameters such as r, N and $1/h$ as expected. Among them, Lagrange polynomials are closest to optimal, followed by the first and the second HS extrapolation functions. In fact, the Lagrange polynomial has advantage over the others in that it is not based on approximation methods; it has demonstrated its high performance by reaching

Chapter V. Conclusion

smaller values of δ than Mosek's limit. On the other hand, though the result that all the extrapolation functions considered are close to optimal for the zero-error model (Appendix E) can be understood in the light of the objective of the optimization problem, it has revealed that our formulation did not work as anticipated. In this respect, we suggest the following directions for future research: to come up with corresponding HS extrapolation functions by referring to Appendix F, which illustrates how the coefficient vector changes with different types error models.

Bibliography

[1] Cirac, J.I. et al. (2021) 'Matrix product states and projected entangled pair states: Concepts, symmetries, theorems', *Reviews of Modern Physics*, 93(4). doi:10.1103/revmodphys.93.045003.

[2] Orús, R. (2014) 'A practical introduction to tensor networks: Matrix product states and projected entangled pair states', *Annals of Physics*, 349, pp. 117–158. doi:10.1016/j.aop.2014.06.013.

[3] Berry, M. (2017) A half-century of physical asymptotics and other diversions, pp. 483–493. doi:10.1142/10480.

[4] Aharonov, Y. et al. (2017) 'The mathematics of Superoscillations', *Memoirs of the American Mathematical Society*, 247(1174), pp. 33–36. doi:10.1090/memo/1174.

[5] Atkinson, K.E. (1988) 'Chapter 1. Mathematical preliminaries', in *An introduction to numerical analysis*. New York, Canada: Wiley, p. 4.

[6] Shilov, G.E. and Silverman, R.A. (1996) *Elementary real and complex analysis*. New York, USA: Dover Publications.

[7] Tchebichef, P. (1867). 'Des valeurs moyennes'. *Journal de Mathématiques Pures et Appliquées*. 2. 12: 177–184.

[8] Exl, L. (2022) 'Numerical Methods III: Optimization', *Lecture notes in Master Computational Science*: University of Vienna.

Bibliography

- [9] Boyd, S.P. and Vandenberghe, L. (2004) Convex optimization. Cambridge, United Kingdom: Cambridge Univ. Pr.
- [10] Bertsekas, D.P. (2009) 'Section 1.10 (Strong convexity)', in Convex optimization theory exercises and Solutions. Nashua (New Hampshire), USA: Athenea Scientific, pp. 13–17.
- [11] Anjos, M.F. and Lasserre, J.B. (2012) '1 Introduction to Semidefinite, Conic and Polynomial Optimization', in Handbook on semidefinite, conic and polynomial optimization. New York: Springer, pp. 1–22.
- [12] Nocedal, J. and Wright, S.J. (2006) 'Numerical optimization'. New York, USA: Springer, pp. 621–622.
- [13] 'Chapter 3. Conic quadratic optimization' (2018) in Mosek Modeling Cookbook. S.I., Denmark: mosek.com, pp. 20–24.
- [14] Borwein, J.M. and Lewis, A.S. (2000) '1.1 Euclidean Spaces', in Convex analysis and nonlinear optimization: Theory and examples. 2nd edn. New York, NY: Springer, pp. 7–8.
- [15] The MathWorks, Inc. (2023). MATLAB version: 9.13.0.2193358 (R2022b Update 5). Available at: <https://www.mathworks.com> (Accessed: 10 November 2023).
- [16] Michael Grant and Stephen Boyd. CVX: Matlab software for disciplined convex programming, version 2.0 beta. <http://cvxr.com/cvx>, September 2013.
- [17] Michael Grant and Stephen Boyd. Graph implementations for nonsmooth convex programs, Recent Advances in Learning and Control (a tribute to M. Vidyasagar), V. Blondel, S. Boyd, and H. Kimura, editors, pages 95-110, Lecture Notes in Control and Information Sciences, Springer, 2008. http://stanford.edu/~boyd/graph_dcp.html.

[18] Grant, M.C. and Boyd, S.P. (2020) CVX Users' Guide. Available at: <http://cvxr.com/cvx/doc/index.html> (Accessed: 10 November 2023).

[19] Sturm, J.F. (2003) Sedumi. Available at: <https://sedumi.ie.lehigh.edu/> (Accessed: 10 November 2023).

[20] Toh, K.C., Tütüncü, R.H. and Todd, M.J. (2001) A MATLAB software package for semidefinite quadratic linear programming, SDPT3. Available at: <https://www.math.cmu.edu/~reha/sdpt3.html> (Accessed: 21 10 November 2023).

[21] Mosek APS (2019) Mosek ApS. Available at: <https://www.mosek.com/> (Accessed: 21 10 November 2023).

[22] Wright, S.J. (1997) Primal-dual interior-point methods. Philadelphia, USA: SIAM.

[23] Cheney, E.W. and Kincaid, D. (2020). Numerical mathematics and computing. Pacific Grove: Brooks/Cole.

[24] Atkinson, K.E. (1988) 'Chapter 3. Interpolation theory', in An introduction to numerical analysis. 2nd edn. New York, Canada: Wiley, pp. 131–134.

[25] Kingma, D.P. and Ba, J. (2017) Adam: A method for stochastic optimization, [1412.6980] Adam: A Method for Stochastic Optimization. Available at: <http://export.arxiv.org/abs/1412.6980> (Accessed: 10 November 2023).

List of Figures

Figure 2-1 Affine (left) and convex (right) sets which have two elements x_1 and x_2 . These can be viewed as affine and convex hulls, respectively. 15

Figure 2-2 Affine (left) and convex (right) sets which have three elements x_1 , x_2 and x_3 . These can be viewed as affine and convex hulls, respectively. 16

Figure 2-3 Convex (left) and nonconvex (right) sets. The two dots and line segments between them illustrate how to tell convexity of sets geometrically; A set C is convex if a line segment connecting any two points in C lies in C as well, otherwise, nonconvex. 16

Figure 2-4 Convex (left) and concave (right) functions. From a position of a line segment between any two points on a graph relative to the function one can determine convexity/concavity of a function graphically. 16

Figure 2-5 [13] Boundaries of quadratic (left) and rotated quadratic (right) cones. 19

Figure 3-1 Change in an optimal solution of the problem (3.1.2) with delta. In the light of the overall appearance of the coefficients, the nonzero terms are named '*sparse coefficients*'. As delta decreases, the number of sparse coefficients increases and so do their absolute values. 29

Figure 3-2 At some delta values, for example $\delta = 0.020$, neighboring non-zero terms show up (marked dark). 30

Figure 3-3 Change in optimal solutions with resolution, obtained using Mosek (*Left column*) and SeDuMi (*Right column*). This illustrates how CVX processes neighboring nonzero components when the resolution ('res') is set lower and thereby increases our trust in the approximation. 30

Figure 3-4 Illustration of how the index-delta plot (Figure 3-5) is constructed out of sparse coefficients obtained over the feasible delta interval. 32

List of Figures

Figure 3-5 Plot of indices of sparse coefficients against delta. We group the points into five; \mathbf{c}_{100} , $\mathbf{m1}$, $\mathbf{m2}$, $\mathbf{m3}$ and $\mathbf{m4}$. Whereas the last term always exists regardless of delta, the others tend to 'move' towards higher indices in the downhill direction of delta. \mathbf{m}^* groups are related through 'metastable intervals' in that whenever $\mathbf{m1}$ enters such an interval, the others start to make their appearance one by one. 32

Figure 3-6 For the ℓ_1 norm of the coefficient vector and \mathbf{m}^* groups, their values against delta are respectively fitted using MATLAB with 95% confidence bounds on coefficients of their fit functions. Details of the fit functions are given in Table 3-2. 33

Figure 3-7 For the ℓ_1 norm of the coefficient vector and \mathbf{m}^* groups, their values against delta less than 0.020 are respectively fitted using MATLAB with 95% confidence bounds on coefficients of their fit functions. Details of the fit functions are given in Table 3-3. Note the linearity of all the fit functions. 35

Figure 3-8 Change in an optimal solution of the problem (3.1.2) with tau. The so-called sparse coefficients are observed as well. Their number and absolute values increase with tau, while the sign of sparse coefficients alternates between plus and minus. Note that, at $\tau = 2.5$, there exist six sparse coefficients in total according to the approximation introduced before. 37

Figure 3-9 Plot of indices of sparse coefficients against tau over the feasible tau interval. Note the appearance of a new index group, $\mathbf{m5}$ 37

Figure 3-10 The index-delta plot over small values of delta, which completes Figure 3-5. Though the numerical solver expects up to six sparse coefficients, as delta approximates zero, more and more nonzero terms are assume to show up to the point the ansatz (3.1.1) rather looks continuous. 38

Figure 4-1 Change in the approximation error with N for the reasonable settings ($T = 1, \tau = 2$ and $E = 1$). Here A and B denote $e^{-iE\tau}$ and $\left(1 - \frac{1}{N}i\tau E\right)^N$, respectively. It indicates that N needs to be sufficiently large to make a reasonable approximation. 43

Figure 4-2 Change in the approximation error with N for the reasonable settings ($\tau = 2$ and $E = 1$). Here A and M denote $e^{-iE\tau}$ and its McLaurin expansion $\sum_{j=0}^N \frac{d^j(e^{-iEt})}{dt^j} \Big|_{t=0} \frac{\tau^j}{j!}$, respectively. 44

Figure 4-3 Change in the approximation error with $1/h$ and j for the reasonable settings ($\tau = 2$ and $E = 1$). Here M and D denote the McLaurin expansion $e^{-iE\tau}$ and its numerical differentiation $\sum_{j=0}^N \frac{d^j(e^{-iE\tau})}{dt^j} |_{t=0} \frac{\tau^j}{j!}$, respectively. 45

List of Tables

Table 2-1 [18] Different capabilities of some solvers CVX supports. Y stands for Yes, N for No, and E for Experimental. 25

Table 3-1 Indices and values of sparse terms associated with the first two plots taken at $\delta = 0.004$ in Figure 3-3. These are accordingly derived from Mosek and SeDuMi as well. Whereas indices of sparse terms coincide, their values are comparable up to three digits. 31

Table 3-2 Details of fit functions in Figure 3-6 including their R-square values. All the numbers are rounded to four significant digits. 34

Table 3-3 Details of fit functions in Figure 3-7 including their R-square values. All the numbers are rounded to four significant digits. 36

Table 4-1 Changes in δ and γ_ε with r for $(T, \tau, E_{max}) = (1, 2, 1)$. Both δ and γ_ε decrease with r 47

Table 4-2 Changes in δ and γ_ε with r for $(T, \tau, E_{max}) = (1, 1.5, 1)$. Overall improvement in the optimality is observed compared to when $\tau = 2$ (Table 4-1). 47

Table 4-3 Changes in δ and γ_ε with r for $(T, \tau, E_{max}) = (1, 2, 2)$. Overall deterioration in the optimality is observed compared to when $E_{max} = 1$ (Table 4-1). 47

Table 4-4 Changes in δ and γ_ε with r for $(T, \tau, E_{max}) = (1, 2, 1)$. Both δ and γ_ε decrease with N 48

Table 4-5 Changes in δ and γ_ε with r for $(T, \tau, E_{max}) = (1, 1.5, 1)$. Overall improvement in the optimality is observed compared to when $\tau = 2$ (Table 4-4). 48

Table 4-6 Changes in δ and γ_ε with r for $(T, \tau, E_{max}) = (1, 2, 2)$. Overall deterioration in the optimality is observed compared to when $E_{max} = 1$ (Table 4-4). 48

Table 4-7 Changes in δ and γ_ε with $1/h$ for $(N, T, \tau, E_{max}) = (6, 1, 2, 1)$. Both δ and γ_ε decrease with $1/h$ 49

List of Tables

Table 4-8 Changes in δ and γ_ε with $1/h$ for $(N, T, \tau, E_{max}) = (6, 1, 1.5, 1)$. An overall worsening in the optimality is observed compared to when $\tau = 2$ (Table 4-7). 49

Table 4-9 Changes in δ and γ_ε with $1/h$ for $(N, T, \tau, E_{max}) = (10, 1, 2, 2)$. Overall deterioration in the optimality is observed compared to when $E_{max} = 1$ (Table 4-7). 49

Appendix A

Justification of the equation (1.4.5)

To estimate $\langle a, f \rangle$ by the procedure of generating a random variable α such that $\langle \alpha \rangle \equiv \langle a, f \rangle$, it is required to choose t and to measure the observable of $A(t)$ on the system. Let a_t be the random variable corresponding to the measurement result and $\mu(t) dt$ the probability distribution for sampling $t \in [0, T]$. The support of $\mu(t)$ must contain that of $f, S(f)$; otherwise, $\langle a, f \rangle$ cannot be evaluated. Once a_t has been identified, one can generate a random variable through the function. This will be denoted as $\alpha := g(a_t, t)$. If $g(a_t, t)$ is deterministic, the only function g such that

$$\langle \alpha \rangle = \int_0^T \mu(t) \langle g(a_t, t) \rangle dt = \int_0^T f(t) \langle a_t \rangle dt = \langle a, f \rangle \quad (\text{A.1})$$

holds for all distributions of a_t is $g(a_t, t) = \frac{f(t)}{\mu(t)} a_t$, for $t \in S(f)$ or 0, otherwise. If g is non-deterministic, the most general function will be of the form $\frac{f(t)}{\mu(t)} a_t + o_t$, where o_t is an independent random variable with $\langle o_t \rangle = 0$. In this case, the final estimator α will have a higher variance than its deterministic counterpart, with $o_t = 0$. There exists, therefore, infinitely many unbiased estimations for $\langle a, f \rangle$, depending on the choice of the distribution $\mu(t)$. Among all those estimators, we wish to identify the one with the minimum variance. We have that

$$\langle \alpha^2 \rangle = \int_{S(f)} \mu(t) \left(\frac{f(t)}{\mu(t)} a_t \right)^2 dt \leq |A|^2 \int_{S(f)} \mu(t) \left(\frac{f(t)}{\mu(t)} \right)^2 dt := |A|^2 A_f(\mu) \quad (\text{A.2})$$

with equality if, for all $t \in S(f)$, $a_t \in \{-|A|, |A|\}$. Hence, in the worst-case scenario, the variance of our estimator α satisfies

$$(\Delta\alpha)^2 := \langle \alpha^2 \rangle - \langle \alpha \rangle^2 = |A|^2 A_f(\mu) - \langle a, f \rangle^2 \quad (\text{A.3})$$

Appendix A. Justification of the equation (1.4.5)

To minimize $\Lambda_f(\mu)$ over all distributions $\mu(t)dt$, consider the average of a non-negative function under $\mu(t)dt$:

$$\begin{aligned}
 & \int_{s(f)} \mu(t) \left(\frac{f(t)}{\mu(t)} - |f(t)|_1 \right)^2 dt \\
 &= \int_{s(f)} \mu(t) \left(\frac{f(t)}{\mu(t)} \right)^2 dt + \int_{s(f)} \mu(t) |f(t)|_1^2 dt - 2|f(t)|_1 \int_{s(f)} |f(t)| dt \\
 &= \Lambda_f(\mu) - \left(1 + \int_{[0,T] \setminus s(f)} \mu(t) dt \right) |f|_1^2
 \end{aligned} \tag{A.4}$$

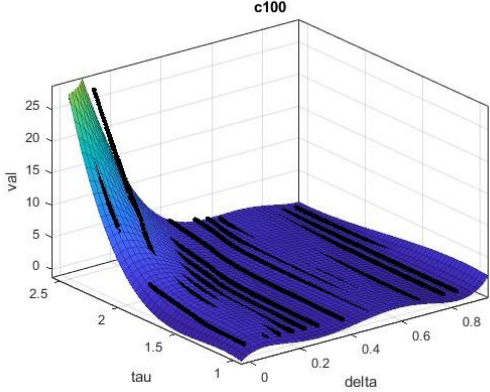
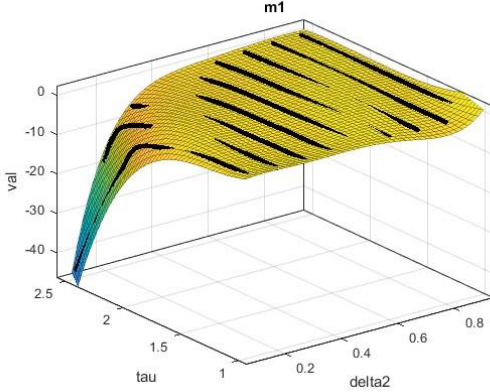
As it is a non-negative number,

$$\Lambda_f(\mu) \geq \left(1 + \int_{[0,T] \setminus s(f)} \mu(t) dt \right) |f|_1^2 \geq |f|_1^2 \tag{A.5}$$

This inequality is saturated when choosing $\mu(t)dt \equiv \frac{|f(t)|}{|f|_1} dt$ as in equation (1.4.5). □

Appendix B

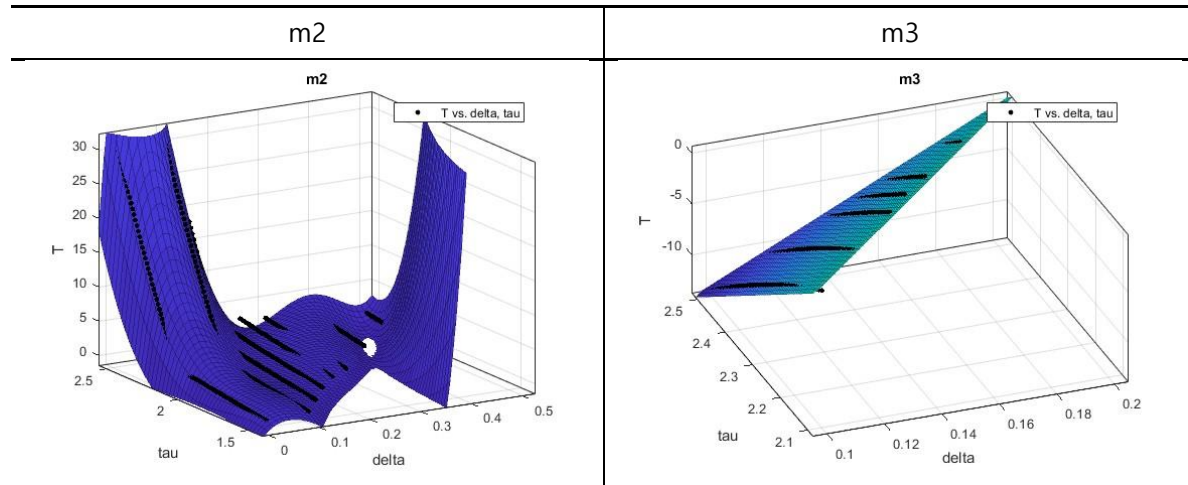
Fit functions of delta and tau

c100	m1
	
<p> $f(x,y) = p00 + p10*x + p01*y + p20*x^2 + p11*x*y + p02*y^2 + p30*x^3 + p21*x^2*y$ with coefficients: p00 = -36.46 (-44.66, -28.27) p10 = 54.81 (7.533, 102.1) p01 = 87.71 (72.88, 102.5) p20 = 4.824 (-83.95, 93.6) p11 = -158.7 (-238.7, -78.59) p02 = -68.57 (-77.24, -59.89) p30 = 348.6 (226.4, 470.7) p21 = -193.8 (-301.3, -86.4) p12 = 177.2 (131.9, 222.4) p03 = 17.78 (16.14, 19.43) p40 = -781.9 (-895, -668.9) p31 = 396.5 (330.4, 462.6) p22 = -51.76 (-102.2, -1.327) p13 = -54.31 (-62.8, -45.81) </p>	<p> $f(x,y) = p00 + p10*x + p01*y + p20*x^2 + p11*x*y + p02*y^2 + p30*x^3 + p21*x^2*y$ with coefficients: p00 = 61.74 (49.78, 73.7) p10 = -81.4 (-148.5, -14.33) p01 = -149.6 (-170.6, -128.5) p20 = -149.5 (-298.8, -0.1132) p11 = 281.4 (177.6, 385.2) p02 = 120.8 (108.6, 132.9) p30 = 298.7 (65.79, 531.5) p21 = 50.2 (-91.75, 192.1) p12 = -274.4 (-331.3, -217.5) p03 = -34.28 (-36.59, -31.98) p40 = -583.4 (-797.2, -369.5) p31 = 247.3 (151.6, 342.9) p22 = -87.72 (-148.6, -26.82) p13 = 112.3 (101.7, 122.9) </p>

Appendix B. Fit functions of delta and tau

p50 = 402.5 (356.9, 448.2)
 p41 = -102.1 (-126.5, -77.67)
 p32 = -75.61 (-89.73, -61.49)
 p23 = 39.82 (30.98, 48.66)
 R-square: 0.9506

p50 = 535.8 (454.1, 617.5)
 p41 = -542.5 (-577.7, -507.2)
 p32 = 293.5 (275.1, 311.8)
 p23 = -86.03 (-96.35, -75.71)
 R-square: 0.9806



$f(x,y) = p00 + p10*x + p01*y + p20*x^2 + p11*x*y + p02*y^2 + p30*x^3 + p21*x^2*y + p12*x*y^2 + p03*y^3 + p40*x^4 + p31*x^3*y + p22*x^2*y^2 + p13*x*y^3 + p50*x^5 + p41*x^4*y + p32*x^3*y^2 + p23*x^2*y^3$

with coefficients:

p00 = -232.4 (-295.9, -169)
 p10 = -437.7 (-1695, 819.4)
 p01 = 421.4 (322.1, 520.7)
 p20 = -5950 (-1.057e+04, -1336)
 p11 = 1157 (-896, 3210)
 p02 = -253.5 (-304.7, -202.4)
 p30 = 1.159e+04 (176.4, 2.301e+04)
 p21 = 5227 (-2886, 1.334e+04)
 p12 = -943.2 (-2044, 157.9)
 p03 = 50.53 (41.87, 59.2)
 p40 = 3.41e+04 (2.465e+04, 4.355e+04)
 p31 = -2.469e+04 (-3.717e+04, -1.221e+04)
 p22 = 670.7 (-4086, 5427)

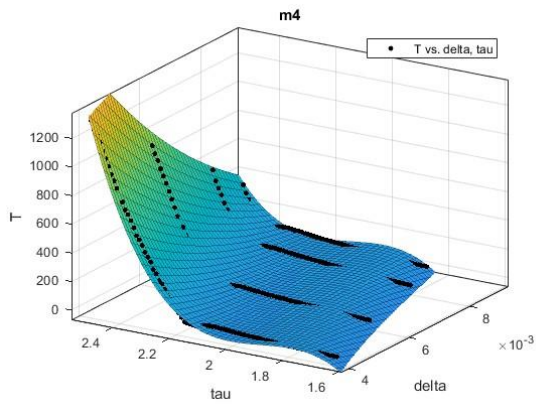
$f(x,y) = p00 + p10*x + p01*y$

with coefficients:

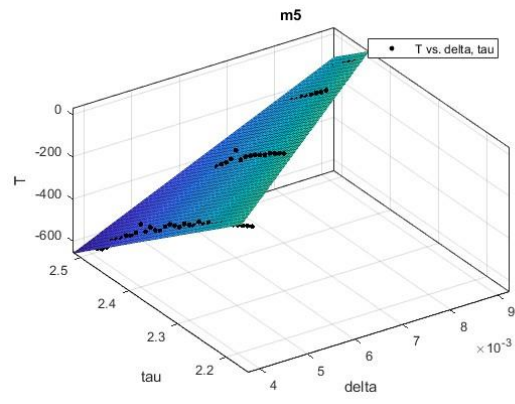
p00 = 56.75 (56.11, 57.4)
 p10 = 129 (127.8, 130.2)
 p01 = -33.02 (-33.32, -32.72)
 R-square: 0.9977

$p_{13} = 264.2$ (70.18, 458.2)
 $p_{50} = 5.456e+04$ (5.115e+04, 5.798e+04)
 $p_{41} = -4.632e+04$ (-5.081e+04, -4.184e+04)
 $p_{32} = 1.507e+04$ (1.171e+04, 1.843e+04)
 $p_{23} = -1358$ (-2276, -440.7)
 R-square: 0.9811

m4



m5



$f(x,y) = p_{00} + p_{10}*x + p_{01}*y + p_{20}*x^2 + p_{11}*x*y + p_{02}*y^2 + p_{30}*x^3 + p_{21}*x^2*y + p_{12}*x*y^2 + p_{03}*y^3$
 with coefficients:
 $p_{00} = -3.05e+04$ (-3.441e+04, -2.66e+04)
 $p_{10} = -2.953e+06$ (-3.31e+06, -2.595e+06)
 $p_{01} = 5.54e+04$ (4.955e+04, 6.125e+04)
 $p_{20} = -6.814e+07$ (-1.018e+08, -3.452e+07)
 $p_{11} = 3.395e+06$ (3.131e+06, 3.658e+06)
 $p_{02} = -3.316e+04$ (-3.61e+04, -3.022e+04)
 $p_{30} = -6.829e+08$ (-2.249e+09, 8.829e+08)
 $p_{21} = 4.096e+07$ (3.259e+07, 4.932e+07)
 $p_{12} = -9.787e+05$ (-1.042e+06, -9.156e+05)
 $p_{03} = 6558$ (6066, 7050)
 R-square: 0.9749

$f(x,y) = p_{00} + p_{10}*x + p_{01}*y$
 with coefficients:
 $p_{00} = 3933$ (3784, 4082)
 $p_{10} = 9.935e+04$ (9.584e+04, 1.029e+05)
 $p_{01} = -1973$ (-2040, -1906)
 R-square: 0.9844

Appendix C

Ideas from duality

Consider the optimization problem (4.1.2):

$$\begin{aligned} & \underset{f, \varepsilon, \nu}{\text{minimize}} \int_0^T |f(t)| \varepsilon(t) dt + \Omega \nu \\ & \text{subject to} \quad |\Delta(E)|^2 \leq \nu, \quad \forall E \in [-E_{max}, E_{max}] \end{aligned} \quad (4.1.2)$$

To ease the notation, any integration in t and E will be hereafter understood to be taken in the interval $[0, T]$ and $[-E_{max}, E_{max}]$, respectively. The Lagrangian of the problem (4.1.2) is:

$$\begin{aligned} \mathcal{L} &= \int \varepsilon(t) |f(t)| dt + \Omega \nu - \int \Lambda(E) dE \left(\nu - \left| \int f(t) e^{-iEt} dt - e^{-iE\tau} \right|^2 \right) \\ &= \int \varepsilon(t) |f(t)| dt + \Omega \nu \\ &\quad - \int \Lambda(E) dE \left(\nu - \int_{T \times T} f(t) f(t') e^{-iE(t-t')} dt dt' - 1 + \int f(t) (e^{-iE(t-\tau)} + e^{iE(t-\tau)}) dt \right) \end{aligned} \quad (C.1)$$

where Λ is a non-negative real function.

The KKT conditions for this problem are:

$$\begin{aligned} & \varepsilon(t) \text{sign}(f(t)) \\ & + \int \Lambda(E) dE \left(\int f(t') (e^{-iE(t-t')} + e^{iE(t-t')}) dt' - (e^{-iE(t-\tau)} + e^{iE(t-\tau)}) \right) = 0, \quad \forall t \in [0, T] \end{aligned} \quad (C.2)$$

$$\Omega - \int \Lambda(E) dE = 0 \quad (C.3)$$

$$\nu \geq \left| \int_0^T f(t) e^{-iEt} dt - e^{-iE\tau} \right|^2, \quad \forall E \in [-E_{max}, E_{max}] \quad (C.4)$$

$$\Lambda(E) \geq 0, \quad \forall E \in [-E_{max}, E_{max}] \quad (C.5)$$

$$\Lambda(E) \left(\nu - \int f(t) f(t') e^{-iE(t-t')} dt dt' - 1 + \int f(t) (e^{-iE(t-\tau)} + e^{iE(t-\tau)}) dt \right) = 0 \quad (C.6)$$

where each instance of 'sign' denotes an arbitrary value in the interval $[-1, 1]$. The first two conditions (C.2) and (C.3) demand the optimality with respect to $f(t)$ and ν ; the next two (C.4) and (C.5) concern primal and dual feasibility, respectively; and the last one (C.6) enforces strong duality.

Appendix C. Ideas from duality

By multiplying (C.2) by $f(t)$ and integrating it, we have

$$\begin{aligned}
& \int \varepsilon(t)|f(t)| dt \\
& + \int \Lambda(E) dE \left(2 \int_{\mathbb{T} \times \mathbb{T}} f(t)f(t') e^{-iE(t-t')} dt dt' - \int f(t)(e^{-iE(t-\tau)} + e^{iE(t-\tau)}) dt \right) \\
& = \int \varepsilon(t)|f(t)| dt + \int \Lambda(E) dE \left(2\nu - 2 + \int f(t)(e^{-iE(t-\tau)} + e^{iE(t-\tau)}) dt \right) \\
& = \int \varepsilon(t)|f(t)| dt + 2(\nu - 1)\Omega + \int \Lambda(E) dE \int f(t)(e^{-iE(t-\tau)} + e^{iE(t-\tau)}) dt
\end{aligned} \tag{C.7}$$

where (C.6) is used in the first quality and (C.3) in the second one.

Though the optimization problem (4.1.2) does not have an independent dual problem, it can be reformulated as

$$\begin{aligned}
& \underset{f, \varepsilon, \mu, \nu}{\text{minimize}} \int \varepsilon(t)\mu(t) dt + \Omega\nu \\
& \text{subject to} \quad \mu(t) - f(t) \geq 0, \quad \mu(t) + f(t) \geq 0, \quad \forall t \\
& \quad \left(\nu, \frac{1}{2}, \int f(t) \cos(Et) dt - \cos(E\tau), \int f(t) \sin(Et) dt - \sin(E\tau) \right) \in K_{sqr} \\
& \quad \forall E \in [-E_{max}, E_{max}]
\end{aligned} \tag{C.8}$$

where $K_{sqr} := \{(u, v, \vec{w}) \in \mathbb{R}_{\geq 0}^2 \times \mathbb{R}^d : 2uv \geq |\vec{w}|^2\}$ is the rotated second-order cone (Definition 2.1.25). Thereby K_{sqr} is self-dual.

Then the Lagrangian of this problem (C.8) is

$$\begin{aligned}
\mathcal{L}' & = \int \varepsilon(t)\mu(t) dt + \Omega\nu - \int (\mu(t) - f(t))V^+ dt - \int (\mu(t) + f(t))V^- dt \\
& - \int \langle Z(E), \left(\nu, \frac{1}{2}, \int f(t) \cos(Et) dt - \cos(E\tau), \int f(t) \sin(Et) dt - \sin(E\tau) \right) \rangle dE
\end{aligned} \tag{C.9}$$

where $Z(E) \in K_{sqr}$ for all $E \in \mathcal{E}$ and V^+ and V^- are non-negative real functions. The angle brackets are used to denote the inner product between the two elements surrounded by themselves, that is, $\langle a, b \rangle := \text{Re}(\langle a|b \rangle)$.

Minimizing the Lagrangian \mathcal{L}' with respect to f, μ and ν and imposing that the minimum is not $-\infty$, we arrive at the dual problem of the primal one (C.8).

$$\begin{aligned}
& \text{maximize}_{v^+, v^-, z, \varepsilon} \int \langle Z(E), (0, \frac{1}{2}, (0, 0, \cos(Et), \sin(Et))) \rangle dE \\
& \text{subject to} \quad \varepsilon(t) - V^+(t) - V^-(t) = 0 \\
& \quad V^+(t) - V^-(t) - \int \langle Z(E), (0, 0, \cos(Et), \sin(Et)) \rangle dE = 0 \quad (\text{C.10}) \\
& \quad \Omega - \int \langle Z(E), (1, 0, 0, 0) \rangle dE = 0 \\
& \quad Z(E) \in K_{\text{Sqrt}}, \quad \forall E \in [-E_{\text{max}}, E_{\text{max}}]
\end{aligned}$$

In addition, strong duality demands that

$$\begin{aligned}
& V^+(t)(\mu(t) - f(t)) = V^-(t)(\mu(t) + f(t)) \\
& = \langle Z(E), \left(v, \frac{1}{2}, \int f(t) \cos(Et) dt - \cos(E\tau), \int f(t) \sin(Et) dt - \sin(E\tau) \right) \rangle = 0 \quad (\text{C.11})
\end{aligned}$$

Alternatively, the optimization problem (4.1.2) can be reformulated in terms of δ :

$$\begin{aligned}
& \text{minimize}_{f, \varepsilon, \mu, v} \int \varepsilon(t)\mu(t) dt + \Omega\delta \\
& \text{subject to} \quad \mu(t) - f(t) \geq 0, \quad \mu(t) + f(t) \geq 0, \quad \forall t \\
& \quad \left(\delta, \int f(t)e^{-iEt} dt - e^{-iE\tau} \right) \in K_{\mathbb{C}}, \quad \forall E \in [-E_{\text{max}}, E_{\text{max}}] \quad (\text{C.12})
\end{aligned}$$

where the complex quadratic cone $K_{\mathbb{C}} := \{(u, \vec{v}) \in \mathbb{R}_{\geq 0} \times \mathbb{C}^d : u \geq |\vec{v}|\}$. Since this cone $K_{\mathbb{C}}$ is also self-dual, the Lagrangian associated with the problem (C.12) is

$$\begin{aligned}
\mathcal{L}'' = & \int \varepsilon(t)\mu(t) dt + \Omega\delta - \int (\mu(t) - f(t))V^+(t) dt - \int (\mu(t) + f(t))V^-(t) dt \\
& - \int \langle Z(E), \left(\delta, \int f(t)e^{-iEt} dt - e^{-iE\tau} \right) \rangle dE \quad (\text{C.13})
\end{aligned}$$

where $Z(E) \in K_{\mathbb{C}}$, for all $E \in \mathcal{E}$ and V^+ and V^- are non-negative real functions.

The dual of the problem (C.12) is therefore

$$\begin{aligned}
& \text{maximize}_{v^+, v^-, z, \varepsilon} \int \langle Z(E), (0, e^{-iE\tau}) \rangle dE \\
& \text{subject to} \quad \varepsilon(t) - V^+(t) - V^-(t) = 0 \\
& \quad V^+(t) - V^-(t) - \int \langle Z(E), (0, e^{-iEt}) \rangle dE = 0 \quad (\text{C.14}) \\
& \quad \Omega - \int \langle Z(E), (1, 0) \rangle dE = 0 \\
& \quad Z(E) \in K_{\mathbb{C}}, \quad \forall E \in [-E_{\text{max}}, E_{\text{max}}]
\end{aligned}$$

Appendix C. Ideas from duality

and strong duality enforces

$$\begin{aligned} V^+(t)(\mu(t) - f(t)) &= V^-(t)(\mu(t) + f(t)) = 0 \\ \langle Z(E), \left(\delta, \int f(t)e^{-iEt} dt - e^{-iE\tau} \right) \rangle &= 0 \end{aligned} \tag{C.15}$$

Appendix D

Error models associated with Section 4.3

(a) Lagrange polynomial

$j \backslash r$	4	5	6	7	8
1	-5.56E-14	2.33E-18	1.05E-14	8.85E-27	1.02E-18
2	3.32E-18	-1.65E-15	1.83E-16	2.27E-19	3.17E-25
3	-2.22E-13	1.92E-18	-4.36E-16	4.24E-27	-2.27E-22
4	1.16E-19	-1.65E-15	2.91E-16	-2.33E-20	1.60E-25
5	-2.21E-13	1.31E-18	-4.95E-16	5.06E-28	-2.53E-21
6	3.10E-18	-1.65E-15	3.86E-16	-1.90E-20	1.44E-27
7		5.49E-19	-4.08E-16	5.24E-27	-3.63E-21
8			4.66E-16	9.16E-21	1.63E-25
9				9.79E-27	7.54E-21
10					3.19E-25

Table D-1 Components of the error model ε for $(T, \tau, E_{max}) = (1, 2, 1)$. This table is associated with Table 4-1.

$j \backslash r$	4	5	6	7	8
1	3.29E-09	3.63E-10	-2.73E-11	4.21E-21	1.45E-09
2	1.86E-09	-4.90E-09	1.11E-16	-7.09E-20	7.79E-19
3	-1.36E-08	8.73E-11	-8.43E-10	1.59E-21	-5.94E-11
4	3.37E-09	-1.07E-08	1.16E-16	-4.93E-19	2.92E-19
5		2.10E-10	-8.43E-10	1.21E-21	-7.49E-11
6			3.24E-16	-4.50E-19	2.20E-19
7				3.88E-21	-3.67E-11
8					7.14E-19

Table D-2 Components of the error model ε for $(T, \tau, E_{max}) = (1, 1.5, 1)$. This table is associated with Table 4-2.

Appendix D. Error models associated with Section 4.3

$j \backslash r$	4	5	6	7	8
1	6.59E-10	2.58E-15	-5.70E-14	6.29E-18	3.62E-13
2	1.58E-10	-1.64E-13	1.60E-17	-1.65E-15	5.49E-20
3	-1.23E-09	1.03E-15	-2.27E-13	3.96E-18	-1.55E-14
4	3.59E-10	-1.65E-13	1.84E-17	-1.65E-15	1.22E-20
5		1.46E-15	-2.26E-13	6.75E-20	-1.73E-14
6			4.16E-17	-1.65E-15	3.42E-20
7				4.06E-18	-1.21E-14
8					6.98E-20

Table D-3 Components of the error model ε for $(T, \tau, E_{max}) = (1, 2, 2)$. This table is associated with Table 4-3.

(b) HS extrapolation function based on superoscillations

$j \backslash N$	10	11	12	13	14
1	1.29E-15	3.02E-14	5.95E-19	1.59E-15	1.14E-26
2	1.24E-10	4.02E-21	1.53E-12	1.60E-25	4.30E-16
3	9.41E-16	-1.19E-17	4.60E-19	2.31E-17	8.84E-27
4	1.61E-12	2.81E-21	1.75E-14	1.23E-25	2.50E-18
5	5.48E-16	-1.67E-17	3.12E-19	1.95E-19	6.14E-27
6	-4.48E-13	1.51E-21	-5.27E-16	8.23E-26	1.15E-20
7	1.35E-16	-1.67E-17	1.56E-19	-1.43E-20	3.27E-27
8	-4.80E-13	1.67E-22	-7.38E-16	3.99E-26	-2.27E-20
9	2.84E-16	-1.67E-17	4.73E-21	-1.98E-20	3.64E-28
10	-3.26E-13	1.18E-21	-7.19E-16	3.34E-27	-2.38E-20
11	6.92E-16	-1.67E-17	1.65E-19	-2.06E-20	2.55E-27
12		2.50E-21	-5.12E-16	4.65E-26	-2.43E-20
13			3.21E-19	-5.93E-22	5.43E-27
14				8.87E-26	-8.10E-21
15					8.18E-27

Table D-4 Components of the error model ε for $(T, \tau, E_{max}) = (1, 2, 1)$. This table is associated with Table 4-4.

$j \backslash N$	11	12	13	14	15
1	9.86E-10	3.35E-15	2.48E-05	1.16E-17	9.07E-07
2	6.20E-19	7.08E-07	1.10E-16	1.45E-08	1.13E-20
3	6.17E-14	2.74E-15	2.75E-08	9.76E-18	6.27E-10
4	4.73E-19	8.09E-10	8.71E-17	1.37E-11	9.27E-21
5	-1.41E-16	2.05E-15	8.01E-11	7.69E-18	1.51E-12
6	3.10E-19	3.99E-11	6.18E-17	3.25E-13	6.99E-21
7	-4.82E-16	1.31E-15	1.54E-12	5.46E-18	4.22E-14
8	1.37E-19	-5.65E-12	3.50E-17	5.98E-15	4.59E-21
9	-5.03E-16	5.28E-16	-1.93E-12	3.12E-18	-2.29E-15
10	3.96E-20	-7.75E-12	7.46E-18	-9.96E-15	2.11E-21
11	-5.02E-16	2.67E-16	-2.05E-12	7.24E-19	-4.55E-15
12	2.15E-19	-6.71E-12	2.03E-17	-1.10E-14	4.09E-22
13		1.05E-15	-1.80E-12	1.69E-18	-4.60E-15
14			4.76E-17	-7.21E-15	2.92E-21
15				4.07E-18	-3.58E-15
16					5.38E-21

Table D-5 Components of the error model ε for $(T, \tau, E_{max}) = (1, 1.5, 1)$. This table is associated with Table 4-5.

Appendix D. Error models associated with Section 4.3

$j \backslash N$	11	12	13	14	15
1	3.01E-14	3.69E-20	1.98E-15	1.78E-26	1.50E-04
2	2.14E-22	1.53E-12	2.27E-26	2.15E-16	1.18E-17
3	-1.21E-17	3.15E-20	2.86E-17	1.17E-26	1.13E-06
4	1.61E-22	1.75E-14	1.97E-26	1.25E-18	1.04E-17
5	-1.67E-17	2.26E-20	1.96E-19	1.47E-26	1.08E-08
6	8.72E-23	-5.26E-16	1.50E-26	5.17E-21	8.26E-18
7	-1.67E-17	1.12E-20	-6.36E-20	1.12E-26	-5.48E-11
8	1.63E-24	-7.38E-16	8.82E-27	-1.19E-20	5.52E-18
9	-1.67E-17	1.31E-21	-7.04E-20	6.92E-27	-3.66E-10
10	8.41E-23	-7.19E-16	1.80E-27	-1.25E-20	2.40E-18
11	-1.67E-17	1.37E-20	-7.14E-20	1.98E-27	-3.76E-10
12	1.58E-22	-5.12E-16	5.38E-27	-1.27E-20	8.93E-19
13		2.47E-20	-4.65E-20	3.10E-27	-3.74E-10
14			1.20E-26	-4.65E-21	4.12E-18
15				7.94E-27	1.12E-10
16					7.06E-18

Table D-6 Components of the error model ε for $(T, \tau, E_{max}) = (1, 2, 2)$. This table is associated with Table 4-6.

(c) HS extrapolation function based on McLaurin expansion

$j \backslash 1/h$	7	8	9	10	11
1	8.92E-17	3.22E-16	1.57E-16	2.44E-15	5.82E-15
2	-4.24E-15	-5.21E-15	-5.71E-15	-5.89E-15	-7.69E-15
3	1.05E-16	3.14E-16	1.72E-16	2.06E-15	4.40E-15
4	-4.19E-15	-5.16E-15	-6.19E-15	-9.12E-15	-1.58E-14
5	1.12E-16	2.86E-16	1.78E-16	1.59E-15	2.83E-15
6	-4.21E-15	-5.24E-15	-5.87E-15	-6.06E-15	-7.30E-15
7	1.10E-16	2.41E-16	1.75E-16	1.07E-15	1.17E-15

Table D-7 Components of the error model ε for $(N, T, \tau, E_{max}) = (6, 1, 2, 1)$. This table is associated with Table 4-7.

$j \backslash 1/h$	9	10	11	12	13
1	1.57E-16	2.44E-15	5.82E-15	6.89E-16	7.51E-16
2	-5.71E-15	-5.89E-15	-7.69E-15	-5.89E-15	-5.88E-15
3	1.72E-16	2.06E-15	4.40E-15	4.84E-16	6.38E-16
4	-6.19E-15	-9.12E-15	-1.58E-14	-6.30E-15	-7.35E-15
5	1.78E-16	1.59E-15	2.83E-15	2.66E-16	5.10E-16
6	-5.87E-15	-6.06E-15	-7.30E-15	-5.65E-15	-6.11E-15
7	1.75E-16	1.07E-15	1.17E-15	4.14E-17	3.70E-16

Table D-8 Components of the error model ε for $(N, T, \tau, E_{max}) = (6, 1, 1.5, 1)$. This table is associated with Table 4-8.

$j \backslash 1/h$	3	4	5	6	7
1	1.76E-21	7.09E-19	7.53E-19	4.89E-19	1.67E-19
2	5.85E-16	7.88E-12	1.64E-10	2.31E-10	2.52E-10
3	8.74E-22	1.25E-19	9.76E-21	2.11E-19	1.19E-19
4	-4.16E-17	-4.78E-13	-1.07E-11	-1.54E-11	-1.78E-11
5	1.35E-21	8.45E-19	7.39E-19	1.57E-19	4.64E-20
6	-5.10E-17	-5.75E-13	-1.33E-11	-1.98E-11	-2.28E-11
7	1.51E-21	7.87E-19	1.04E-18	4.58E-19	3.57E-20
8	-5.29E-17	-5.95E-13	-1.33E-11	-1.93E-11	-2.19E-11
9	6.37E-22	6.09E-21	7.10E-19	5.63E-19	1.10E-19
10	1.53E-16	1.33E-12	3.20E-11	4.90E-11	5.98E-11
11	1.81E-21	7.81E-19	5.10E-20	4.27E-19	1.62E-19

Table D-9 Components of the error model ε for $(N, T, \tau, E_{max}) = (10, 1, 2, 2)$. This table is associated with Table 4-9.

Appendix E

Ideas for further extrapolation functions

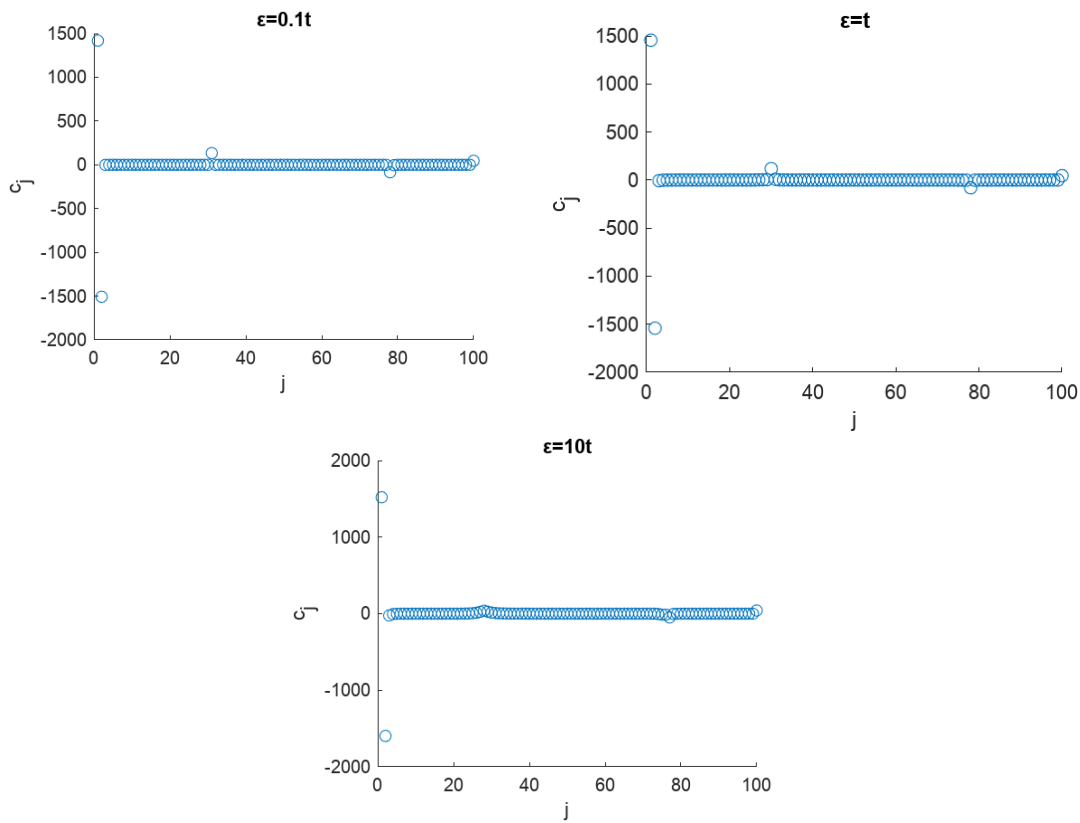
By adopting the ansatz (3.1.1), the optimization problem (4.1.2) is transformed into

$$\begin{aligned} & \text{minimize}_c \sum_{j=1}^n \varepsilon(t) |c_j| \\ & \text{subject to} \left| \sum_{j=1}^n c_j e^{-iEt_j} - e^{-iE\tau} \right| \leq \delta, \quad \forall E \in \mathbb{E} \end{aligned} \tag{E.1}$$

where $c := (c_1, \dots, c_n)$ is the coefficient vector of the ansatz (3.1.1), $t_j := \frac{j-1}{r-1}T$ are time steps with $j = 1, \dots, n$ and $\mathbb{E} := \left\{ \left(-1 + 2 \frac{l-1}{r-1} \right) E_{max} : l = 1, \dots, r \right\}$ for the maximum observable time T , maximum energy E_{max} and resolution r .

For the set of parameters $(\delta, T, \tau, E_{max}, r) = (0.004, 1, 2, 1, 100)$, we solve the optimization problem by using Mosek supported by CVX as before. In this regard, we consider three types of error models: an error model which linearly increases with time multiplied by a constant $\varepsilon_1(t) := at$, an error model which linearly increases with time plus a constant $\varepsilon_2(t) := t + a$, and power models $\varepsilon_3(t)$ in the form of $t^{\frac{1}{3}}$, $t^{\frac{1}{2}}$, and t^2 . Here we use three numbers — 0.1, 1 and 10 — for the scalar parameter a . All the results come with the status message 'Solved'.

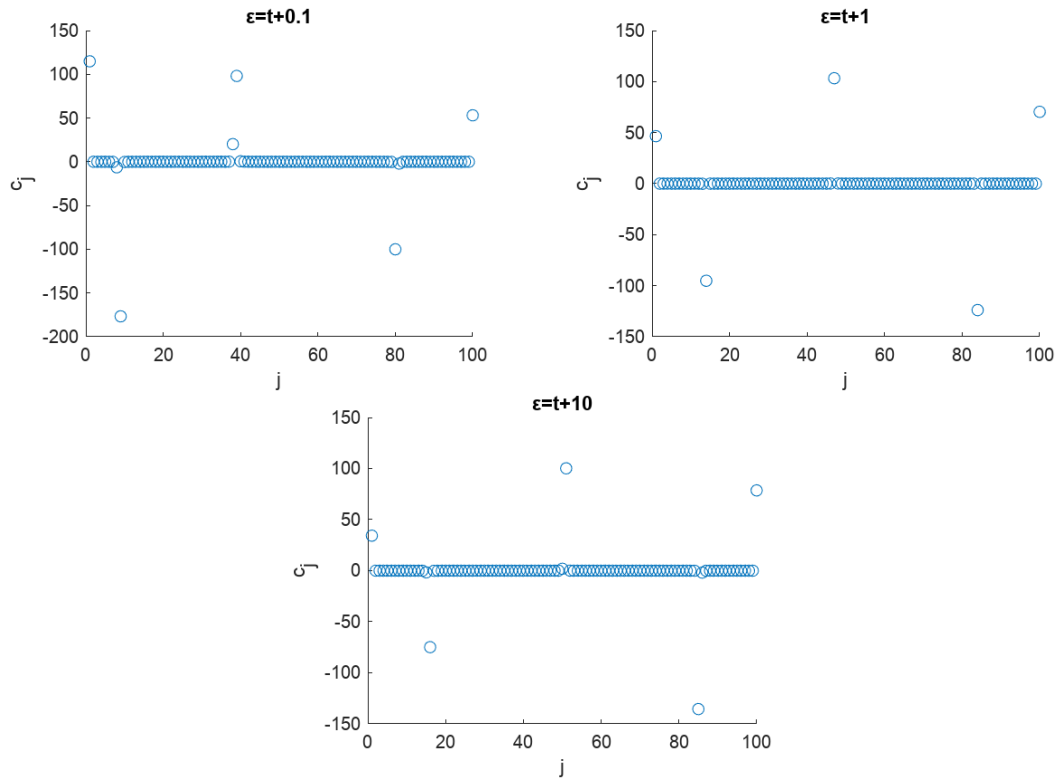
(a) Model linearly increasing with time $\varepsilon_1(t) := at$



$a = 0.1$					
index	1	2	31	78	100
value	1417.830	-1507.490	134.134	-85.124	45.769
$a = 1$					
index	1	2	30	78	100
value	1455.970	-1542.660	118.434	-80.376	45.020
$a = 10$					
index	1	2	28	77	100
value	-1519.560	-1595.830	38.325	-49.362	41.428

Figure & Table E-1 Plots of the coefficient vector c , followed by indices and values of its sparse terms for each error model linearly increasing with time.

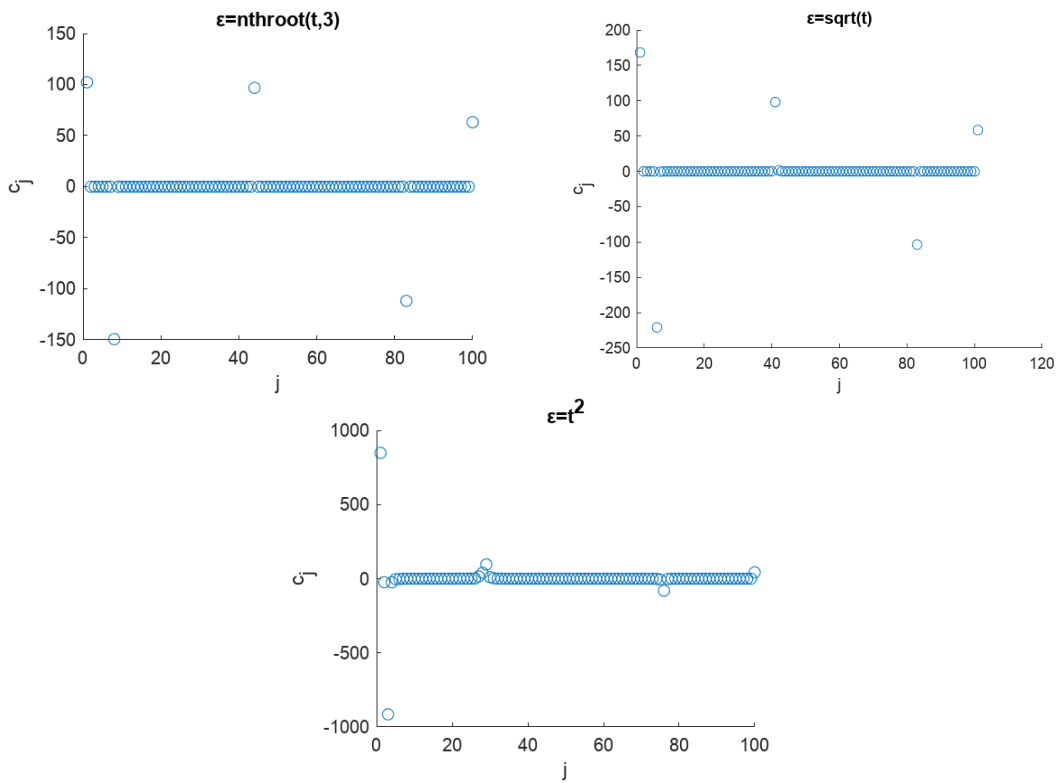
(b) Model linearly increasing with time plus a constant $\varepsilon_2(t) := t + a$



$a = 0.1$							
index	1	8	9	38	39	80	100
value	114.898	-6.342	-176.794	20.157	98.152	53.122	-100.142
$a = 1$							
index	1	14	47	84	100		
value	45.518	-95.236	103.319	-123.985	70.400		
$a = 10$							
index	1	16	51	85	100		
value	34.293	-74.873	100.222	-135.504	78.605		

Figure & Table E-2 Plots of the coefficient vector c , followed by indices and values of its sparse terms for each error model linearly increasing with time with a constant added.

(c) Power models $\varepsilon_3(t)$



$\varepsilon_{3,1}(t) = t^{\frac{1}{3}}$							
index	1	8	44	83	100		
value	102.240	-149.331	96.787	96.787	63.134		
$\varepsilon_{3,2}(t) = t^{\frac{1}{2}}$							
index	1	6	41	82	100		
value	165.125	-217.114	99.096	-104.117	58.117		
$\varepsilon_{3,3}(t) = t^2$							
index	1	3	27	28	29	76	100
value	848.028	-914.810	15.004	39.993	95.713	-81.132	42.890

Figure & Table E-3 Plots of the coefficient vector c , followed by indices and values of its sparse terms for each power error model.