

KAPITEL 5

Kalman-Filtered Compressive-Sensing

5.1 Einleitung

Aus dem Kapitel 4 wissen wir, wenn ein Signal dünnbesetzt ist, kann man das Compressive-Sensing (CS) anwenden, um das Signal durch weniger Messungen zu rekonstruieren, als nach dem Nyquist-Shannon-Theorem gefordert sind. Das CS ist jedoch nicht für nicht-dünnbesetzte Signale geeignet. Daher muss man ein neues Verfahren aufstellen, um das CS anwenden zu können. Durch die Kombination vom CS mit dem Systemmodell (Kalman-Filter) kann man das neue Verfahren realisieren. Im Folgenden werden wir die Temperaturverteilung mit punktförmigen Anregungen aus Kapitel 2 als das Beispiel zur Simulation und Bewertung unseres Algorithmus nehmen.

5.2 Stand der Technik

Es gibt bereits einige Forscher, die an der Kombination vom Compressive-Sensing und dem Kalman-Filter oder anderen Verfahren arbeiten [45]. Z. B.

M. Salman Asif [46; 47] hat den Begriff Homotopie-Parameter in das Compressive-Sensing eingeführt, um eine schnellere und genauere Lösungsmethode des CS zu finden, nämlich die dynamische Aktualisierung für ℓ_1 -Minimierung [47]. Als Anwendungsbeispiel hat er die Methode auf Streaming-Signale und Videos angewendet [46].

S. Ji. et. al. [51] haben Bayes'sche Compressive-Sensing vorgeschlagen, wobei die Messungen adaptiv ausgewählt werden und die Varianz der Fehler auch gleichzeitig geschätzt werden. Dadurch ist der Fehler deutlich kleiner, besonders bei wenigen Messungen.

Im Jahr 2008 hat N. Vaswani [48] erstmals ein Verfahren für die Kombination von Kalman-Filter und Compressive-Sensing entwickelt, bei dem das Compressive-Sensing in der Innovationsschritt (Filterschritt) des Kalman-Filters durchgeführt wird. Das Kalman-Filter schätzt nur die Zustandsgröße in "Support-Set". Ein Support-Set ist die Menge der Indices C , für die \underline{x}_C die Nicht-Null-Elemente sind. Wenn der Fehler größer als der Schwellwert ist, wird das Compressive-Sensing benutzt, um die Support-Set zu aktualisieren. Im [49] hat N. Vaswani das Verfahren verbessert. Das verbesserte Verfahren wird als modifiziertes KFCS bezeichnet, bei dem ein vorläufiges Kalman-Filter eingeführt wird, um die Rechenaufwand zu reduzieren. Das Verfahren funktioniert besonders gut für die sich langsam veränderte Signale.

Aber ihre Arbeit basiert immer auf der Dünnbesetzung des Signals [52] und versucht, die Genauigkeit zu verbessern und die Algorithmen schneller zu machen. Es gibt noch keine Arbeit sich damit beschäftigt, die Anforderung der Dünnbesetzung durch die Kombination des Systemmodells abzuschwächen.

5.3 Kalman-Filter mit zufälligen Messungen

Beim CS macht man zufällige Messungen. Deswegen kombinieren wir zuerst die zufällige Messungen mit dem Kalman-Filter, um eine bessere Schätzung zu erhalten.

5.3.1 Systemmodell

Bevor das CS und das KF kombiniert werden, probieren wir zunächst das Kalman-Filter mit zufälligen Messungen, d. h. die Messwerte werden nicht an festen Stellen gewählt, sondern sich zufällig verändern. Um diese Funktion zu realisieren, können wir vorab viele Sensoren ausbringen, und zu jedem Zeitpunkt aktivieren wir einige Sensoren davon zufällig.

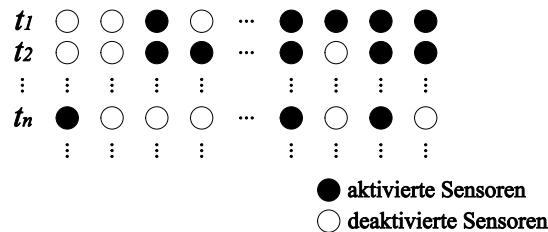


Abbildung 5.1: Eingestellte Sensoren.

Es kann durch eine Delta-Matrix Φ_k beschrieben werden, welche Sensoren zum Zeitpunkt k aktiviert oder deaktiviert werden. Dann können wir das Systemmodell und Messmodell für das rsKF (Engl. random sampling Kalman Filter) herleiten:

- Bei der Modalanalyse gilt

$$\begin{aligned}\underline{T}_{k+1} &= \mathbf{A} \cdot \underline{T}_k + \underline{w}_k , \\ \underbrace{\Phi_{k+1} \cdot \underline{y}_{k+1}}_{=: \tilde{y}_{k+1}} &= \underbrace{\Phi_{k+1} \cdot \Psi}_{=: \tilde{\Psi}} \cdot \underline{T}_{k+1} + \underbrace{\Phi_{k+1} \cdot \underline{v}_{k+1}}_{=: \tilde{v}_{k+1}} ,\end{aligned}$$

also

$$\begin{aligned}\underline{T}_{k+1} &= \mathbf{A} \cdot \underline{T}_k + \underline{w}_k , \\ \underline{y}_{k+1} &= \tilde{\Psi}_{k+1} \cdot \underline{T}_{k+1} + \underline{v}_{k+1} ,\end{aligned}$$

wobei

$$\begin{aligned}\mathbf{C}_{k+1}^{\tilde{v}} &= E \left\{ \underline{v}_{k+1} \cdot \underline{v}_{k+1}^T \right\} \\ &= E \left\{ \Phi_{k+1} \underline{v} \cdot \underline{v}^T \Phi_{k+1}^T \right\} \\ &= \Phi_{k+1} \mathbf{C}_{k+1}^v \Phi_{k+1}^T .\end{aligned}$$

- Bei dem Finite-Differenzen-Verfahren gilt

$$\begin{aligned}\underline{f}_{k+1} &= \mathcal{D} \cdot \underline{f}_k + \underline{w}_k , \\ \underline{y}_{k+1} &= \tilde{\Phi}_{k+1} \cdot \underline{f}_{k+1} + \underline{v}_{k+1} .\end{aligned}$$

Wir nennen \underline{y}_{k+1} die komprimierte Messung.

5.3.2 Simulation

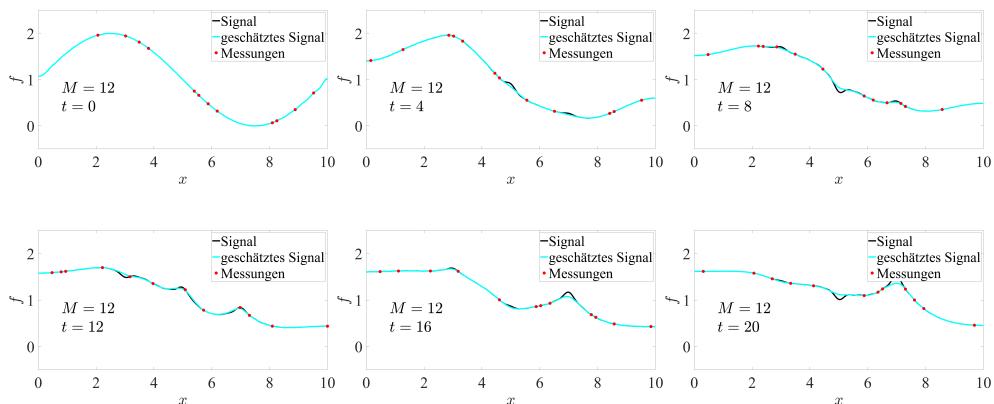


Abbildung 5.2: rsKF aus der FDM mit 12 zufällige Messungen.

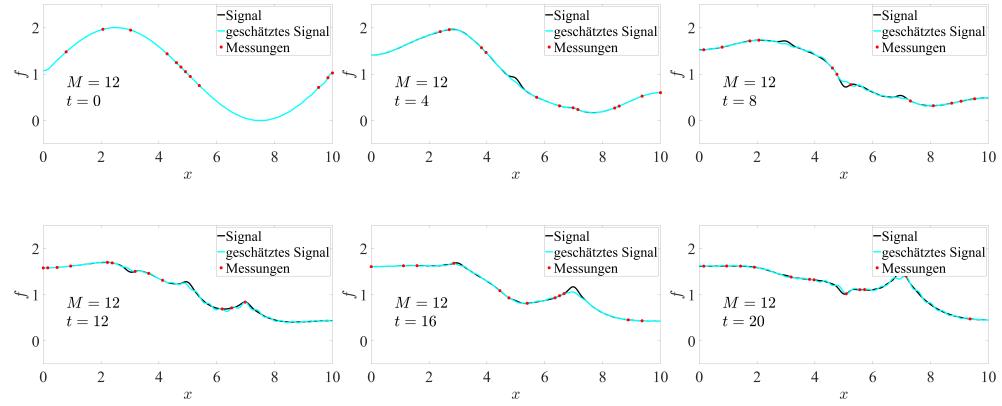


Abbildung 5.3: rsKF aus der Modalanalyse mit 12 zufällige Messungen.

Es ist anschaulich, dass es viele Schwingungen in der Simulation aus der Modalanalyse gibt, weil der Grad G der Modalanalyse größer ist als die Anzahl der Messungen. Wenn der Grad G jedoch zu klein gewählt wird, führt dies dazu, dass das Modell zu der Temperaturverteilung nicht gut passen kann. Daher ist die Wahl des richtigen Grads ein wichtiges Problem. Außerdem ist die FDM, wie in Kapitel 1 dargelegt, breiter verwendet als die Modalanalyse, sodass wir uns in den zukünftigen Diskussionen auf der FDM konzentrieren werden.

5.3.3 Diskussion

Der Vergleich zwischen dem rsKF und dem traditionellen Kalman-Filter (mit festen Sensorstellen) wird in der Abbildung 5.4 gezeigt.

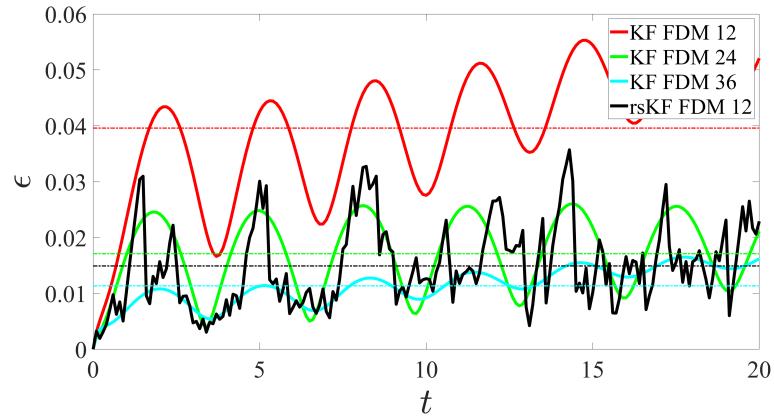


Abbildung 5.4: Vergleich rsKF und KF.

Das Ergebnis aus dem rsKF mit 12 Messungen ist viel besser als das traditionellen KF mit 12 Sensoren. Es ist sogar besser als das KF mit 24 Sensoren. Aber der rsKF hat noch viele Nachteile. Z. B. er wird in einigen Fällen die Position der

Spitzen nicht gemessen, was zu großen Fehlern führen kann, wie in der Abbildung 5.5 gezeigt. Deshalb wollen wir die Genauigkeit der Zustandsschätzung durch die Kombination vom rsKF und dem CS weiter verbessern.

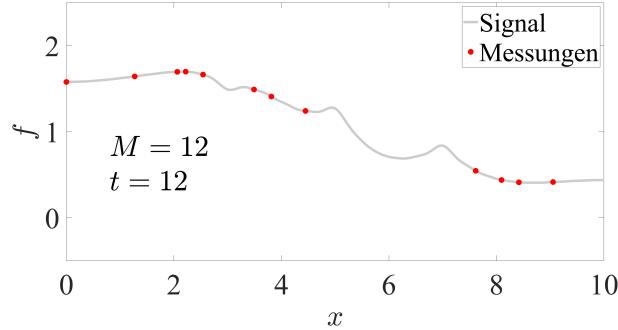


Abbildung 5.5: Nachteil des rsKF, keine Messung fällt an den Spitzen.

5.4 Signal-Analyse

Wie in Kapitel 3 dargelegt, sollten, wenn wir das Compressive-Sensing verwenden wollen, die geschätzten Koeffizienten dünnbesetzt oder komprimierbar sein und gleichzeitig die Basis $\tilde{\Theta}$ die RIP erfüllen muss. Deswegen müssen wir zuerst die Eigenschaften des Signals analysieren, und wenn das Signal nicht komprimierbar ist, müssen wir das Signal durch einige Methoden so dünnbesetzt wie möglich machen, um das Compressive-Sensing verwenden zu können. Und danach treffen wir einige Maßnahmen, um die RIP von $\tilde{\Theta}$ zu garantieren.

5.4.1 Hauptprobleme

Zuerst wird die Komprimierbarkeit des Signals betrachtet. Die beste Methode, um festzustellen, ob ein Signal komprimierbar ist, ist die K -Term-Approximation. Wenn ein Signal mit K Termen approximiert werden kann, bedeutet das, dass das Signal durch ein K -dünnbesetztes Signal approximiert werden kann, mithilfe des Compressive-Sensing können wir dieses K -dünnbesetzte Signal berechnen. Da ε in der Formel (4.2) die Differenz zwischen dem approximierten Signal und dem Originalsignal darstellt, ist es nötig zu beachten, dass ε bei exakt dünnbesetzten Signalen gleich 0 sein kann, da das approximierte Signal genau das Originalsignal ist. Wenn das Signal jedoch nur komprimierbar ist, gibt es immer einen Fehler zwischen dem approximierten Signal und dem ursprünglichen Signal, sodass ε nicht zu klein sein darf. Wenn ε zu klein ist, führt dies zu einer Überanpassung.

Im Folgenden wählen wir $K = 10$, weil wir 12 Messungen zur Rekonstruktion der

Temperaturverteilung verwenden wollen. Daher muss K mindestens kleiner als 12 sein. Außerdem wählen wir den Zeitpunkt $t = 20$ als Beispiel:

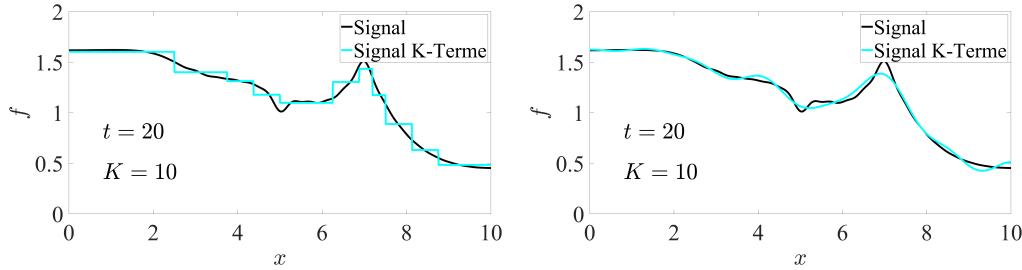


Abbildung 5.6: K -Terme-Approximation links: Waveletbasis, rechts: DCT-Basis.

Von der Abbildung 5.6 können wir ausgehen, dass das Signal nicht K -dünnbesetzt ist, weder mit der Wavelet-Basis noch mit der DCT-Basis. Trotzdem versuchen wir, dieses Signal mit dem Compressive-Sensing zu rekonstruieren.

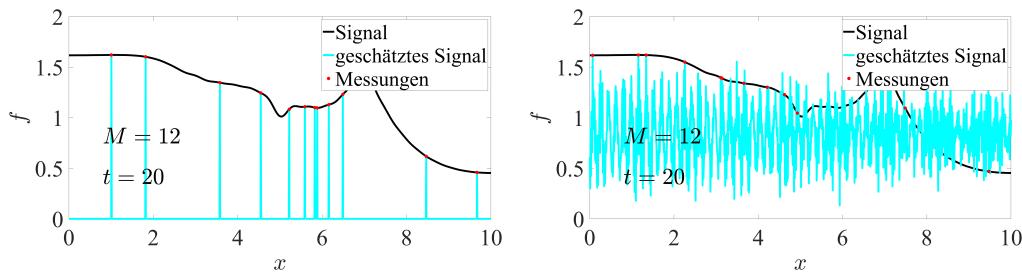


Abbildung 5.7: Rekonstruiertes Signal links: Waveletbasis, rechts: DCT-Basis.

Es ist aus der Abbildung 5.7 anschaulich, dass das rekonstruierte Signal aus dem Compressive-Sensing zu der K -Terme-Approximation überhaupt nicht passt. Der Grund dafür ist, dass die Basis die RIP nicht erfüllt.

Zusammenfassen, wir haben zwei Hauptprobleme getroffen. Erstens, das Signal nicht dünnbesetzt oder komprimierbar ist. Zweitens, die Basis erfüllt die RIP nicht. Im Folgenden werden wir auf diese zwei Probleme fokussieren.

5.4.2 Komprimierbare Änderungen

In dem Beispiel können die manuell gegebenen Anfangsbedingungen beliebig sein, aber nachdem die Anfangsbedingung festgelegt ist, ist die Wärmeleitung mit dem Systemmodell konsistent. So spekuliert man unweigerlich, dass vielleicht die Koeffizienten z_{k+1} selbst nicht komprimierbar ist (weil die Anfangsbedingung beliebig gegeben sein kann), die Änderung der Koeffizienten $\Delta z_k := z_{k+1} - z_k$ jedoch komprimierbar sein könnte, weil die Änderung des Signals einem bestimmten Gesetz

(dem Systemmodell) entspricht. Daher werden wir die Änderung der Koeffizienten betrachten. Folgende Grafik zeigt die K -Terme-Approximation der Änderung der Koeffizienten Δz mit $\Delta t = 0.5$ s, nämlich $t_1 = 19.5$ s und $t_2 = 20$ s, d. h. die Koeffizienten zum Zeitpunkt $t_1 = 19.5$ s wird als bekannt angenommen, während die Koeffizienten zum Zeitpunkt $t_2 = 20$ s zu schätzen sind.

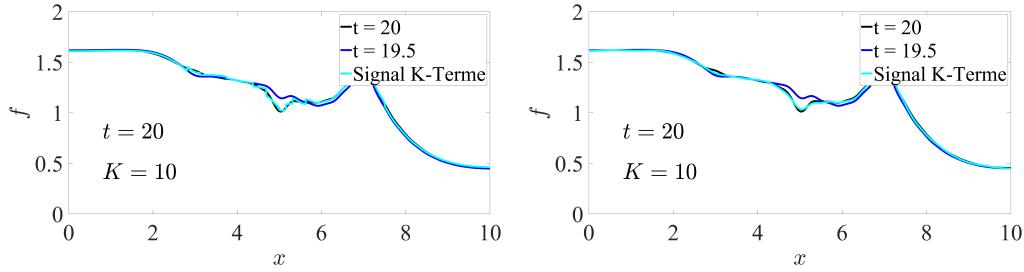


Abbildung 5.8: K -Terme-Approximation für die Änderung der Koeffizienten links: Waveletbasis, rechts: DCT-Basis.

Wie wir aus der obigen Grafik ersehen können, sind die Ergebnisse der K -Terme-Approximation von der Änderung der Koeffizienten viel besser als die von der Koeffizienten selbst, insbesondere mit DCT-Basis. Daher können wir die Änderungen der Koeffizienten als komprimierbar bezeichnen. Aber wir dürfen eine weitere Voraussetzung für das Compressive-Sensing nicht außer Acht lassen, nämlich dass die RIP erfüllt sein muss. Zuerst versuchen wir durch das Compressive-Sensing das Signal zu rekonstruieren, und wir erhalten die Ergebnisse in der Abbildung 5.9.

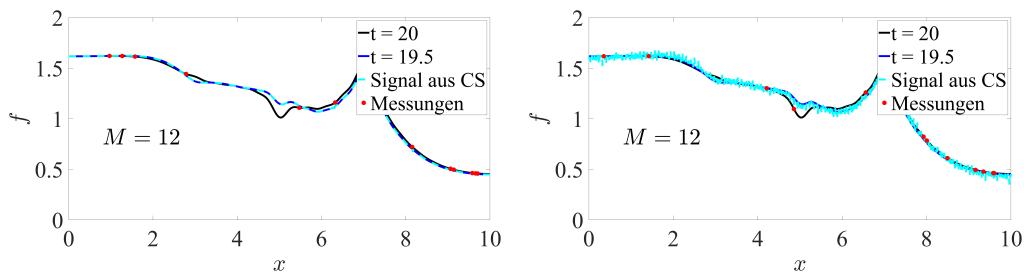


Abbildung 5.9: Rekonstruiertes Signal für die Änderung der Koeffizienten links: Waveletbasis, rechts: DCT-Basis.

Leider können wir durch die ℓ_1 -Norm-Optimierung die richtigen Ergebnisse nicht erzielen. Denn obwohl die Änderung der Koeffizienten Δz komprimierbar ist, wird die andere Bedingung des Compressive-Sensing, die RIP, nicht erfüllt, sodass wir das Problem der RIP noch lösen müssen. Im nächsten Abschnitt werden wir uns auf die Lösung des RIP-Problems konzentrieren, damit wir es ermöglichen, das Kalman-Filter und das Compressive-Sensing zu kombinieren.

5.4.3 Dimensionreduzierung

Aus der Definition der RIP

$$(1 - \delta_K) \|\tilde{z}\|_2^2 \leq \left\| \tilde{\Theta} \cdot \underline{z} \right\|_2^2 \leq (1 + \delta_K) \|\underline{z}\|_2^2 ,$$

wissen wir, dass die RIP den Grad der Orthogonalität der Basis $\tilde{\Theta} \in \mathbb{R}^{M \times N}$ unter K -dünnbesetzten Signalen garantiert. Mit anderen Worten, je kleiner die $\frac{M}{N}$ ist, desto kleiner Möglichkeit es gibt, dass die Basis $\tilde{\Theta}$ die RIP erfüllt. Das heißt, wir sollten N so weit wie möglich reduzieren, um eine höhere Wahrscheinlichkeit zu haben, dass die RIP erfüllt. Deswegen rekonstruieren wir nicht alle Werte an den Knoten, sondern nur eine bestimmte Anzahl der Werte, die ausreichend für die Schätzung der Zustandsgrößen ist. Beispielsweise können wir die Werte nur an den möglichen Sensorstellen rekonstruieren, somit reduziert sich N in diesem Fall von $N = 1024$ auf $N = 64$. Auf diese Weise simulieren wir nun die Fälle im letzten Kapitel:

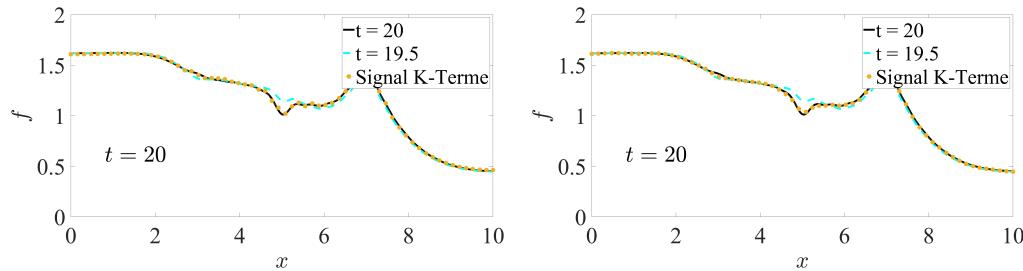


Abbildung 5.10: K -Terme-Approximation für die Änderung der Koeffizienten mit $N = 64$ links: Waveletbasis, rechts: DCT-Basis.

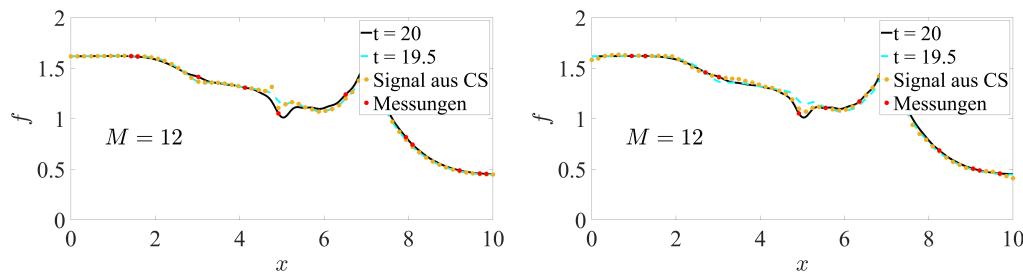


Abbildung 5.11: Rekonstruiertes Signal für die Änderung der Koeffizienten mit $N = 64$ links: Waveletbasis, rechts: DCT-Basis.

Wie aus Abbildung 5.11 ersichtlich ist, haben wir nach der Schätzung von der Änderung der Koeffizienten und der Reduzierung der Dimension relativ genauer Ergebnis erhalten, und das Ergebnis mit der DCT-Basis ist im Vergleich zu dem Ergebnis mit Wavelet-Basis besser. Aus der Modalanalyse können wir auch erkennen, dass die Temperaturverteilung durch einige glatte Cosinusfunktionen gut approximiert werden kann. Die Haar-Wavelets sind jedoch schärfer und mit vielen Sprüngen.

5.4.4 Sparse-Coding-Representation

Aus dem obigen Beispiel ist ersichtlich, dass verschiedene Basen großen Einfluss auf die Ergebnisse haben, da das Signal mit verschiedenen Basen eine unterschiedliche Dünnsbesetzung aufweist. Deshalb hoffen wir, eine Basis zu finden, mit der das Signal so dünnbesetzt wie möglich dargestellt werden kann.

Sparse-Coding-Representation (SCR) war in den letzten Jahren ein großes Thema im Bereich des maschinellen Lernens [34; 35; 36]. Das Ziel der SCR besteht darin, den größten Teil oder das gesamte ursprüngliche Signal in einer Linearkombination aus weniger Basissignalen darzustellen. Die SCR wurde erstmals im Jahr 1996 von Olshausen und Field vorgeschlagen [33], um die visuelle Verarbeitung im Gehirn zu erklären. Wir können die SCR verwenden, um eine Basis zu finden, auf der das Signal dünnbesetzt dargestellt werden kann. Eine Realisierung des SCR-Problems ist Lösung des Optimierungsproblems (5.1).

$$\min_{\underline{\alpha}_i, \Theta} \sum_{i=1}^K \left\{ \left\| \underline{y}_i - \Theta \cdot \underline{\alpha}_i \right\|_2^2 + \lambda_i \|\underline{\alpha}_i\|_0 \right\}$$

NB $\Theta^T \Theta = \mathbf{I}$

(5.1)

Der erste Term steht für den Fehler der Approximation von der Basis Θ und seiner Koeffizienten $\underline{\alpha}_i$ zu allen Zeitpunkten $i = 1, \dots, K$. Der zweite Term garantiert die Dünnsbesetzung der Koeffizienten $\underline{\alpha}_i$ zu jedem Zeitpunkt und die Nebenbedingung bedeutet die Orthogonalität der Basis Θ . Die Abbildung 5.12 zeigt eine Visualisierung der Basis Θ :

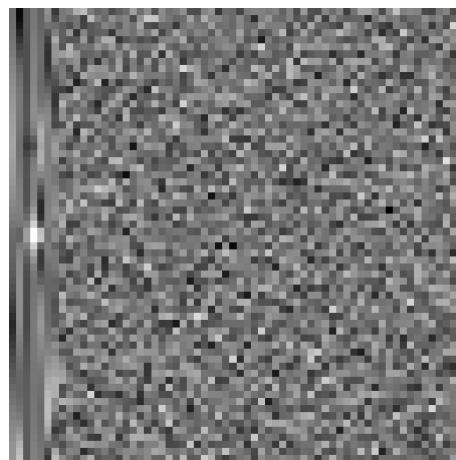


Abbildung 5.12: Visualisierung von Θ .

In der Abbildung 5.12 beschreibt jede Spalte eine Basisfunktion $\underline{\theta}_i$.

5.5 Kalman-Gleichung mit Compressive-Sensing

Um die Informationen aus dem Kalman-Filter und aus dem Compressive-Sensing zu fusionieren, brauchen wir dazu noch ein Verfahren. Zunächst stellen wir eine weitverbreitete Methode vor, die die ℓ_1 -Norm-Optimierung direkt in der Kalman-Gleichung einsetzen.

Die Kalman-Gleichung, z. B. aus der Modalanalyse, ist

$$\underline{T}_{k+1}^e = \arg \min_{\underline{T}_{k+1}} \left\{ \frac{1}{\sigma_S^2} \left\| \mathbf{A} \cdot \underline{T}_k - \underline{T}_{k+1} \right\|_2^2 + \frac{1}{\sigma_M^2} \left\| \tilde{\underline{y}}_{k+1} - \Psi \cdot \underline{T}_{k+1} \right\|_2^2 \right\}, \quad (5.2)$$

wobei σ_S für die Varianz der Systemrauschen und σ_M für die Varianz der Messunsicherheit stehen. Bei der FDM kann man die Zustandsgrößen und die Systemmatrix entsprechend wechseln. Für das Compressive-Sensing gilt

$$\tilde{\underline{y}}_{k+1} = \tilde{\Theta} \cdot \underline{z}_{k+1},$$

und

$$\begin{aligned} \underline{z}_{k+1}^e &= \arg \min_{\underline{z}_{k+1}} \left\| \underline{z}_{k+1} \right\|_1, \\ \text{NB} \quad \left\| \tilde{\underline{y}}_{k+1} - \tilde{\Theta} \cdot \underline{z}_{k+1} \right\|_2^2 &\leq \varepsilon. \end{aligned}$$

Man kann die oben genannten zwei Formeln auch in eine Formel umschreiben [24], also

$$\underline{z}_{k+1}^e = \arg \min_{\underline{z}_{k+1}} \left\{ \left\| \underline{z} \right\|_1 + \left\| \tilde{\underline{y}}_{k+1} - \tilde{\Theta} \cdot \underline{z}_{k+1} \right\|_2^2 \right\}. \quad (5.3)$$

Es ist zu betrachten, dass

$$\begin{aligned} \tilde{\Psi} \cdot \underline{T}_{k+1}^e &= \tilde{\Theta} \cdot \underline{z}_{k+1}^e \\ \underline{T}_{k+1}^e &\approx \left(\tilde{\Psi}^T \tilde{\Psi} \right)^{-1} \tilde{\Psi}^T \tilde{\Theta} \cdot \underline{z}_{k+1}^e \\ &= \left(\tilde{\Psi}^T \tilde{\Psi} \right)^{-1} \tilde{\Psi}^T \tilde{\Theta} \cdot (\underline{z}_k^e + \Delta \underline{z}_k^e) \end{aligned}$$

gilt, wobei $\tilde{\Psi}$ die komprimierende Messabbildung in der Modalanalyse und $\tilde{\Theta}$ die komprimierende Messabbildung im Compressive-Sensing sind. Dann kombiniert man

(5.2) und (5.3) in Form von

$$\begin{aligned} \Delta \underline{z}_k^e &= \arg \min_{\Delta \underline{z}_k} \left\{ G_1 \|\Delta \underline{z}_k\|_1 + G_2 \left(\frac{1}{\sigma_S^2} \left\| \mathbf{A} \cdot \underline{T}_k - (\tilde{\Psi}^T \tilde{\Psi})^{-1} \tilde{\Psi}^T \tilde{\Theta} \right. \right. \right. \\ &\quad \left. \left. \left. \cdot (\underline{z}_k + \Delta \underline{z}_k) \right\|_2^2 + \frac{1}{\sigma_M^2} \left\| \tilde{y}_{k+1} - \tilde{\Theta} \cdot (\underline{z}_k + \Delta \underline{z}_k) \right\|_2^2 \right) \right\} \\ &= \arg \min_{\Delta \underline{z}_k} \left\{ G_1 \|\Delta \underline{z}_k\|_1 + G_2 \left\| \begin{bmatrix} \frac{1}{\sigma_S} \left(\mathbf{A} \cdot \underline{T}_k - (\tilde{\Psi}^T \tilde{\Psi})^{-1} \tilde{\Psi}^T \tilde{\Theta} \cdot (\underline{z}_k + \Delta \underline{z}_k) \right) \\ \frac{1}{\sigma_M} \left(\tilde{y}_{k+1} - \tilde{\Theta} \cdot (\underline{z}_k + \Delta \underline{z}_k) \right) \end{bmatrix} \right\|_2^2 \right\}. \end{aligned}$$

G_1 und G_2 bezeichnen die Gewichtungen für den ℓ_p -Norm-Term und den zweiten Term. Bei den Simulationen werden mehrerer Werte von $\frac{G_2}{G_1}$ gewählt, und zur Vereinfachung wählen wir $\sigma_S = \sigma_M = 1$.

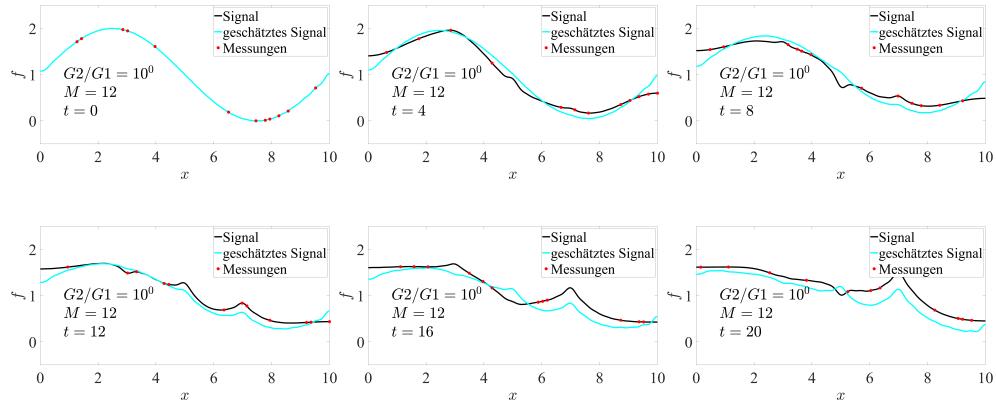


Abbildung 5.13: Rekonstruiertes Signal mit Kalman-Gleichung mit $G_2/G_1 = 10^0$.

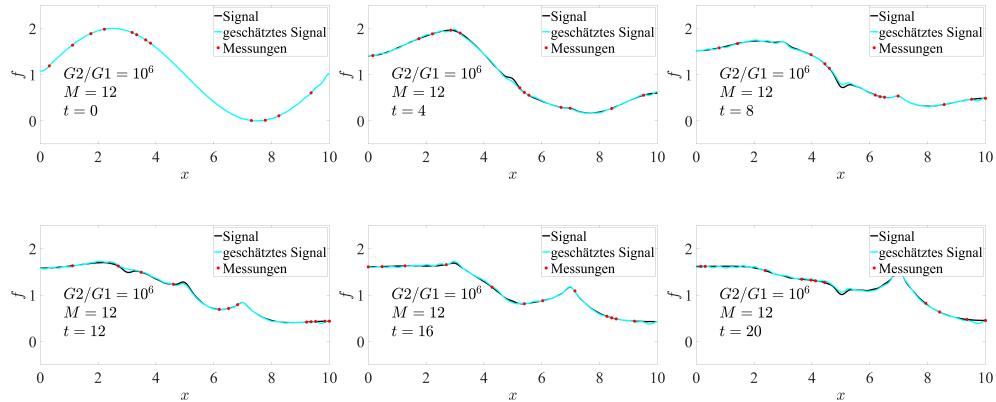


Abbildung 5.14: Rekonstruiertes Signal mit Kalman-Gleichung mit $G_2/G_1 = 10^6$.

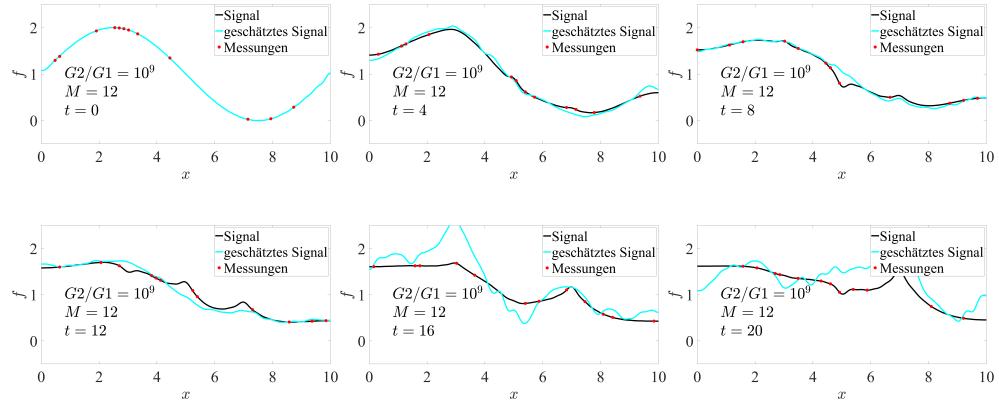


Abbildung 5.15: Rekonstruiertes Signal mit Kalman-Gleichung mit $G_2/G_1 = 10^9$.

Wir können davon ausgehen, dass die Gewichtungen G_1 und G_2 das Ergebnis stark beeinflussen. Um die Wirksamkeit der Gewichtungen zu analysieren, haben wir mehrere Simulationen gemacht, und danach die Fehler bewertet.

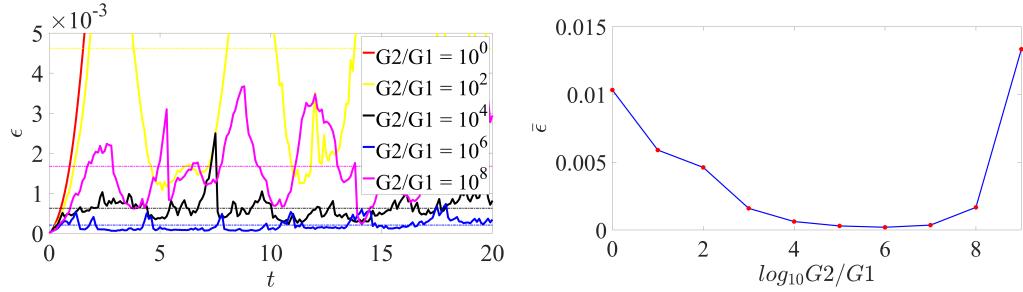


Abbildung 5.16: Fehler mit verschiedenen G_2/G_1 .

Es ist anschaulich, wenn G_2/G_1 zu groß oder zu klein ist, ist der Fehler sehr groß. Das beste Ergebnis kommt bei $G_2/G_1 \approx 10^6$ vor. Aber in der Praxis kennen wir jedoch das echte Signal nicht, sodass es schwierig ist, die Gewichtung G_1 und G_2 zu bestimmen. Deshalb brauchen wir ein anderes Verfahren, bei dem die Forderung auf der Gewichtung abschwächt wird.

5.6 Kalman-Filter mit Pseudo-Messungen

Das Kalman-Filter mit Pseudo-Messungen bedeutet, dass man das CS vor dem KF läuft, dann werden die geschätzte aus dem CS als die Messungen im Kalman-Filter einsetzen. In diesen Ansatz spielen die Gewichtungen G_1 und G_2 wenige Rolle.

5.6.1 Pseudo-Messungen

Im Gegensatz zur vorherigen Methode lassen wir bei dieser Methode zunächst das Compressive-Sensing unabhängig vom KF laufen, wie in der Abbildung 5.17 gezeigt.

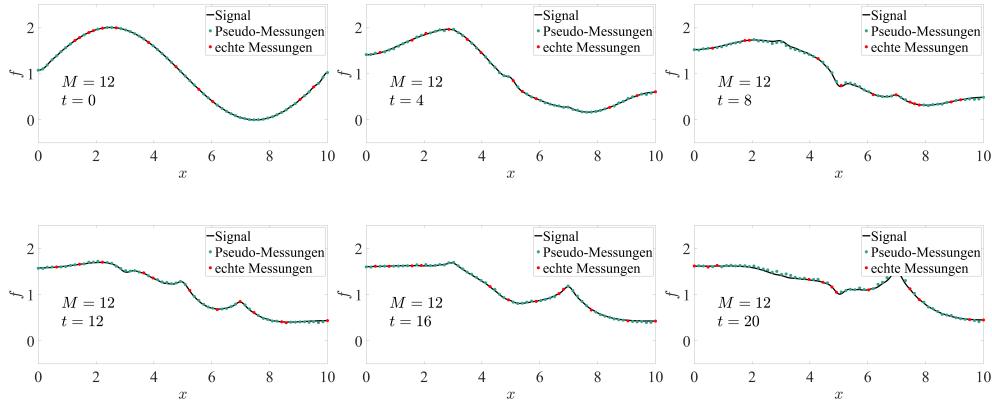


Abbildung 5.17: Pseudo-Messungen.

Diese rekonstruierte Punkte geben wir dann als die Pseudo-Messungen (PM) in das Kalman-Filter ein, und auf diese Weise können wir das Compressive-Sensing und das Kalman-Filter kombinieren.

5.6.2 Dynamische Gewichtung

Auf diese Weise können wir den Einfluss der Gewichtung reduzieren. Aber trotzdem können wir die Messungen noch verbessern. Da die Pseudo-Messungen keine reale Messungen sind, ist die Unsicherheit von Pseudo-Messungen σ_{PM} größer als die reale Messungen σ_M . Deshalb können wir den Pseudo-Messungen höhere Messunsicherheit geben. Es ist auch offenbar, wenn $\sigma_{PM} \rightarrow \infty$, ist das Kalman-Filter mit Pseudo-Messungen gleich wie das rsKF, d. h. nur die tatsächliche Messungen verwendet werden.

Wie wir aus der Abbildung 5.18 ersehen können, ist das Ergebnis nach der Einführung von dynamischer Gewichtung deutlich verbessert, vor allem an den Spitzen. Bemerkenswert ist, dass das Ergebnis an einigen Stellen nach der Einführung dynamischer Gewichtung sogar noch schlechter ist, d. h. es gilt nicht immer, je kleiner die Gewichtung der Pseudo-Messungen eingestellt wird, desto besser wird das Ergebnis. Wir werden später noch über die Messunsicherheit der PM diskutieren.

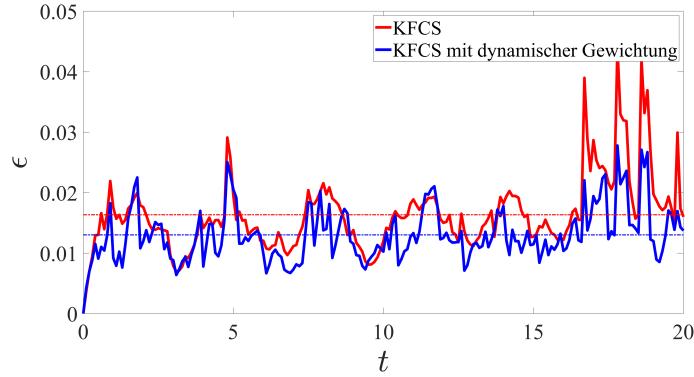


Abbildung 5.18: Vergleich ohne und mit dynamischer Gewichtung.

5.6.3 Fehler-Akkumulation

Da wir nur die Änderung der Koeffizienten Δz schätzen und die Messungen zufällig gewählt werden, führt dies dazu, dass der Fehler sich akkumuliert, wie in der Abbildung 5.19 dargestellt. Wir finden, dass dies daran liegt, dass wir nur die ℓ_1 -Norm-Optimierung vom Compressive-Sensing separat durchführen und die Pseudo-Messungen in das Kalman-Filter setzen. In diesem Prozess sind das Compressive-Sensing und das Kalman-Filter nicht vollständig fusioniert, d. h. die Ergebnisse vom CS wirken auf das KF, aber die Ergebnisse vom KF beeinflussen das CS nicht. Deshalb hoffen wir, dass das KF auch zu einer besserer Abschätzung vom CS beitragen wird. Um dies zu erreichen, können wir die Koeffizientenaktualisierung verwenden.

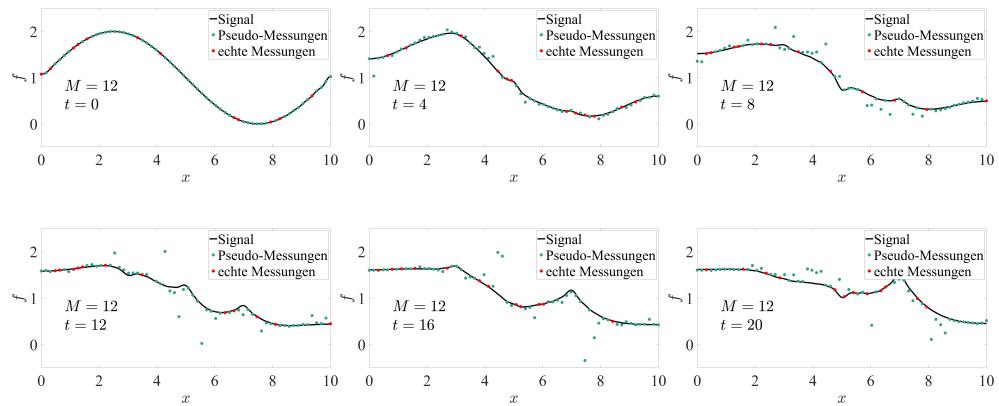


Abbildung 5.19: Pseudo-Messungen mit Wavelet-Basis.

5.7 Koeffizientenaktualisierung

Wie in der Abbildung 5.20 gezeigt, schätzen wir im vorherigen Prozess die Änderung der Koeffizienten Δz_k aus den zufälligen Messungen \tilde{y}_{k+1} mithilfe des Compressive-

Sensing, dann berechnen wir $\underline{z}_{k+1} = \underline{z}_k + \Delta \underline{z}_k$. Mit \underline{z}_{k+1} rekonstruieren wir die Pseudo-Messungen y_{K+1}^{PM} . Dann setzen wir die Pseudo-Messungen in das Kalman-Filter ein, um den aktuellen Zustandsgröße T_{k+1} zu schätzen, schließlich berechnen wir die Temperaturverteilung f (bei der FDM ist die Zustandsgröße direkt f).

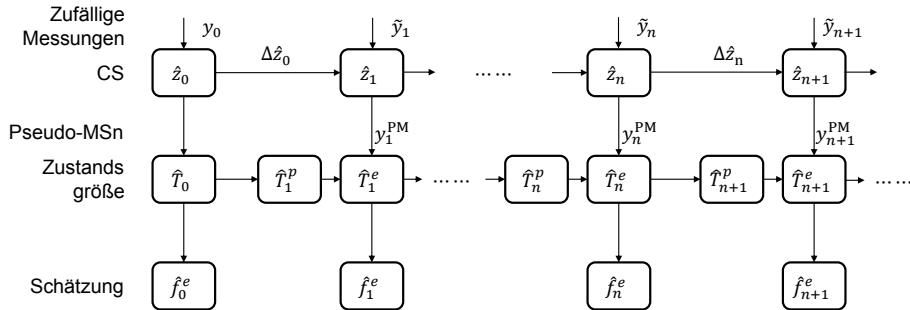


Abbildung 5.20: Verlauf des vorherigen KFCS.

Wie wir aus dem obenstehenden Diagramm ersehen können, ist der Grund für die Fehlerakkumulation darin zu sehen, dass es keine Rückkopplung im Prozess gibt, weshalb wir eine Rückkopplung einführen sollten, die es der Ausgangsgröße f ermöglicht, auf den Eingang (Compressive-Sensing) zu wirken.

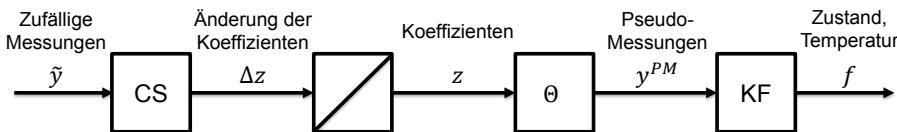


Abbildung 5.21: Verlauf des vorherigen KFCS.

Ausgehend von dieser Idee können wir den Prozess in die folgende Struktur 5.22 ändern

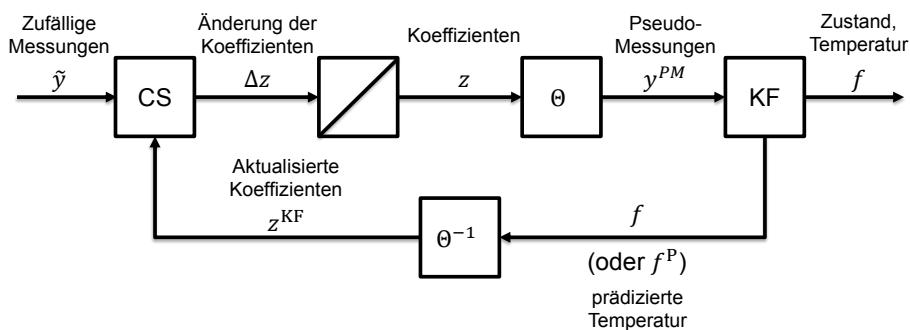


Abbildung 5.22: Strukturbild des KFCS mit Rückkopplung.

Dazu gibt es zwei Möglichkeiten: Koeffizientenaktualisierung aus der Schätzung und aus der Prädiktion.

5.7.1 Aktualisierung aus der Schätzung

Die erste Möglichkeit ist, wie in der Abbildung 5.23 gezeigt, dass wir die Temperaturverteilung \underline{f}_k nach der Schätzung der Zustandsgröße \underline{T}_k oder \underline{f}_k vom Kalman-Filter berechnen, und dann berechnen wir die Koeffizienten $\underline{z}_k^{\text{KF}}$ aus der Temperaturverteilung \underline{f}_k auf der Basis Θ . Dann verwenden wir $\underline{z}_k^{\text{KF}}$ als den Anfangswert, um $\Delta \underline{z}_k$ zu berechnen, wobei $\underline{z}_{k+1} = \underline{z}_k^{\text{KF}} + \Delta \underline{z}_k$.

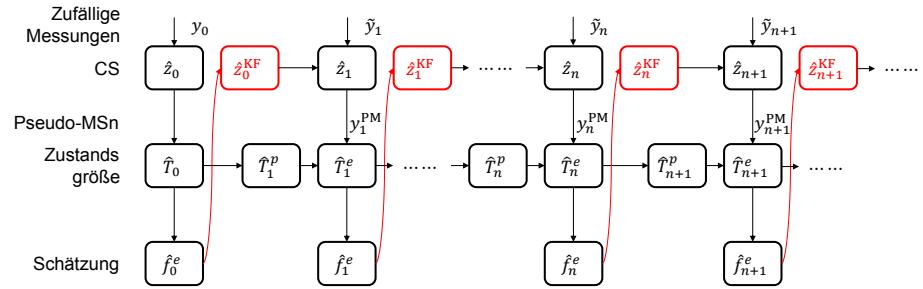


Abbildung 5.23: Verlauf für Koeffizientenaktualisierung aus der Schätzung.

Dieser Ansatz ist eher auf Fälle anwendbar, in denen die Anregung unbekannt ist. Für den Fall, dass die Anregung bekannt ist, können wir die Koeffizienten \underline{z}_k mit der Temperaturverteilung aus der Prädiktion aktualisieren, um bessere Ergebnisse zu erhalten.

5.7.2 Aktualisierung aus der Prädiktion

Die zweite Möglichkeit ist, wie in der Abbildung 5.24 gezeigt, dass wir die Temperaturverteilung \underline{f}_k aus der Prädiktion der Zustandsgröße \underline{T}_k^p oder \underline{f}_k^p vom Kalman-Filter berechnen, und dann berechnen wir die Koeffizienten \underline{z}_k^p aus der prädizierte Temperaturverteilung auf der Basis Θ . Dann verwenden wir \underline{z}_k^p als den Anfangswert, um $\Delta \underline{z}_k$ zu berechnen, wobei $\underline{z}_{k+1} = \underline{z}_k^p + \Delta \underline{z}_k$.

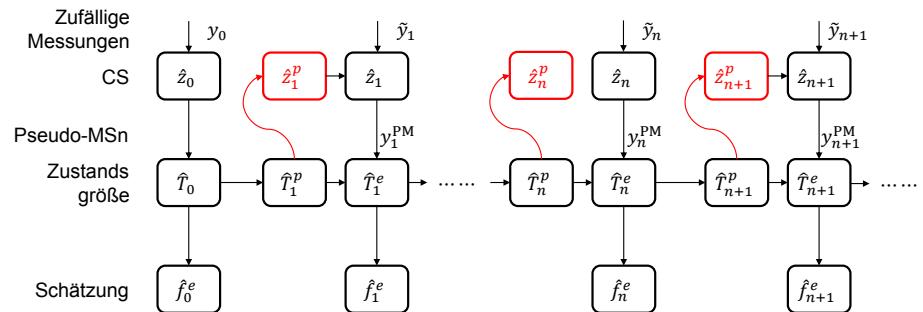


Abbildung 5.24: Verlauf für Koeffizientenaktualisierung aus der Prädiktion.

5.8 Iteratives Verfahren

Da dieses Verfahren nichtlinear ist, können wir auch ein iteratives Verfahren verwenden, um die Schätzung zu verbessern. Zu jedem Zeitpunkt k berechnen wir die Zustandsgrößen $(\underline{T}_k)^i$ und die Koeffizienten $(\underline{z}_k)^i$ mit einer Beobachtung \tilde{y}_k und Prädiktion \hat{T}_k^P iterativ, bis sie konvergieren oder die maximale Anzahl von Iterationen erreicht ist. Die konvergierenden Koeffizienten $(\underline{z}_k)^i$ werden als die \underline{z}_k^{KF} zur Berechnung der $\Delta \underline{z}_{k+1}$ zum nächsten Zeitpunkt verwendet. Das Verfahren ist wie in der folgenden Grafik 5.25 gezeigt:

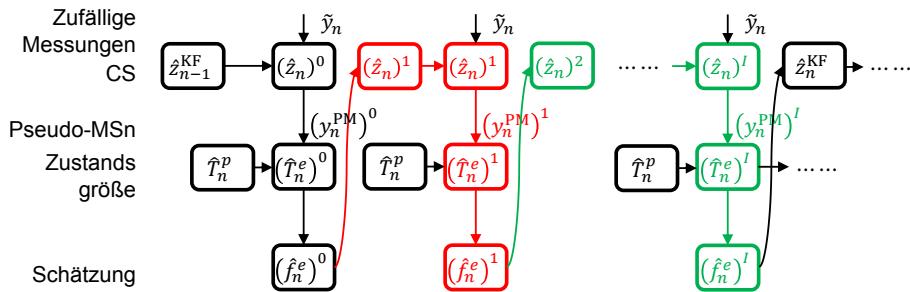


Abbildung 5.25: Verlauf für iteratives Verfahren.

5.9 Simulation

Wir rekonstruieren die Temperaturverteilung im vorherigen Beispiel mit verschiedenen Verfahren, nämlich dem Kalman-Filter mit 12, 40 und 64 äquidistanten Sensoren, das rsKF mit 12 Messungen mit 64 verfügbaren Sensoren, das KFCS (Kalman-Filter mit Pseudo-Messungen ohne Rückkopplung) mit 12 Messungen mit 64 verfügbaren Sensoren, und das iterative KFCS mit 12 Messungen mit 64 verfügbaren Sensoren.

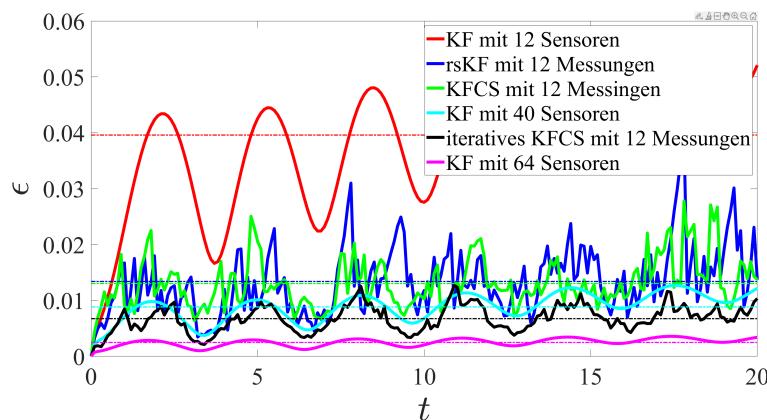


Abbildung 5.26: Vergleich aller Verfahren.

Wir können aus der Abbildung 5.26 sehen, dass das Ergebnis von rsKF mit 12 Messungen und KFCS mit 12 Messungen ähnlich sind. Der Grund ist, wie wir im Abschnitt 5.3.2 gesagt haben, wenn die dynamische Gewichtung der Pseudo-Messungen zu niedrig gewählt werden, sind die beiden Methoden gleichwertig. Aber nur, weil wir die Pseudo-Messungen eingeführt und die Struktur von CS-KF-CS aufgebaut haben, konnte das Verfahren im Vergleich zu rsKF noch weiter verbessert werden. Als Endergebnis können wir sehen, dass iteratives KFCS mit 12 Messungen bereits besseres Ergebnis liefert als das traditionelle Kalman-Filter mit 40 Sensoren. Das bedeutet, dass das neue Verfahren in der Arbeit sehr effektiv ist.

5.10 Zusammenfassung

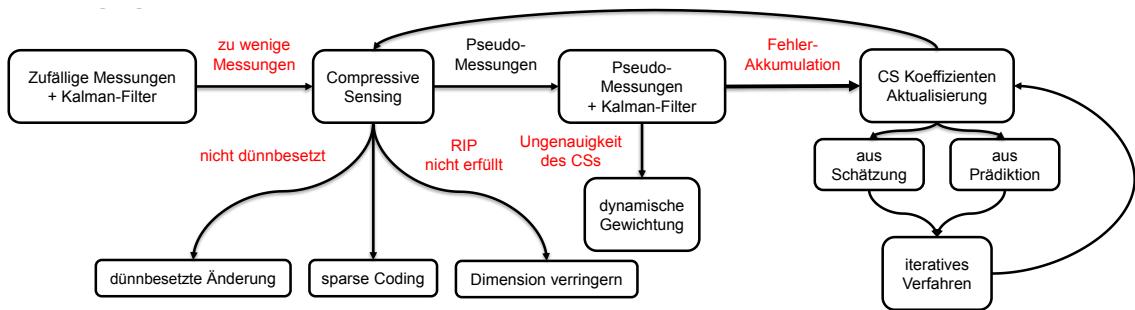


Abbildung 5.27: Vollständiger Gedankengang zu diesem Kapitel.

In diesem Kapitel untersuchten wir zunächst das Kalman-Filter mit zufälligen Messungen (rsKF). Und wir fanden den Nachteil von rsKF: Es ist möglich, dass es an einigen Stellen keine Messpunkte gibt. Deswegen wollen wir das Kalman-Filter und das Compressive-Sensing, also die zwei Vorwissen: das Systemmodell und die Dünnbesetzung des Signals kombinieren. In diesem Fall haben wir zwei Ansätze versucht: die Kombination der Kalman-Gleichung mit der ℓ_p -Norm-Optimierung und die Kombination vom Compressive-Sensing und dem Kalman-Filter mittels Pseudo-Messungen. Tatsächlich müssen wir im ersten Ansatz die Gewichte der beiden Teile bestimmen, was in dem Fall, dass das echte Signal unbekannt ist, sehr schwierig ist. Deshalb verfolgen wir hauptsächlich den zweiten Ansatz. Beim zweiten Ansatz besteht jedoch eine hohe Anforderung an die Dünnbesetzung des Signals, und das Ergebnis ist bei nicht-dünnbesetzten Signalen oft sehr schlecht, d. h. die Pseudo-Messungen haben wenigen Sinn. Dazu werden drei Verfahren vorgeschlagen, um die Anforderungen an die Dünnbesetzung des Signals abzuschwächen. Darüber hinaus schlagen wir zwei zusätzliche Methoden vor, um die Fusion der beiden Vorwissen zu verbessern und damit die Genauigkeit der Schätzung weiter zu erhöhen. Zunächst richten wir eine Rückkoppelung ein, d. h. wir aktualisieren die Koeffizienten \underline{z} aus

dem Ergebnis des Kalman-Filters (Schätzung oder Prädiktion) als Anfangswert vom Compressive-Sensing. Zweitens führen wir ein iteratives Verfahren ein, bei dem wir die Koeffizienten zu jedem Zeitpunkt iterativ berechnen, bis sie konvergieren. Durch diese Verfahren können wir mit 12 Messungen ein besseres Ergebnis im Vergleich zu traditionellem Kalman-Filter mit 40 Messungen erreichen.

