Masterarbeit

Approximation der Dynamik eines epidemiologischen Modells im hierarchischen Tuckerformat

Benedikt Schröter
Studiengang: M.Sc. Informatik
Matrikelnummer: 7487868

08.11.2023

Betreuer: PD Dr. Arne Nägel

Erklärung zur Abschlussarbeit

gemäß § 34, Abs. 16 der Ordnung für den Masterstudiengang Informatik vom 17. Juni 2019

Hiermit erkläre ich

(Nachname, Vorname)

Die vorliegende Arbeit habe ich selbstständig und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel verfasst.
Ebenso bestätige ich, dass diese Arbeit nicht, auch nicht auszugsweise, für eine andere Prüfung oder Studienleistung verwendet wurde.
Zudem versichere ich, dass die von mir eingereichten schriftlichen gebundenen Versionen meiner Masterarbeit mit der eingereichten elektronischen Version meiner Masterarbeit übereinstimmen.

Frankfurt am Main, den

Unterschrift der/des Studierenden

Danksagung

Mein erster Dank gilt meinem Betreuer Herrn PD Dr. Arne Nägel, der sich immer Zeit für meine Anliegen genommen hat. Dafür, dass ich jede Lagebesprechung mit einem guten Gefühl und frischer Motivation verlassen konnte, möchte ich mich ausdrücklich bedanken.

Ein besonderer Dank geht auch an meine Freundin Öykü, die nicht nur immer ein offenes Ohr für mich hatte, sondern mir vor allem in der finalen Phase der Arbeit den Rücken freigehalten hat.

Meinem Bruder Flo danke ich dafür, dass er sich über jeden kleinen Fortschritt während der Arbeit mit mir gefreut hat. Bei meiner Schwester Lena möchte ich mich herzlich für das gewissenhafte Korrekturlesen meiner Arbeit bedanken.

Der abschließende Dank gebührt meinem Vater, der mich über mein ganzes Studium hinweg unterstützt hat. Ohne deine Hilfe wäre das Studium in dieser Form nicht möglich gewesen.

Inhaltsverzeichnis

1 Einleitung

Der Begriff *curse of dimensionality* geht auf den Mathematiker Richard Bellman zurück und bezieht sich auf verschiedene Herausforderungen im Umgang mit Daten aus hochdimensionalen Räumen. Die zugrundeliegende Problematik besteht darin, dass das Volumen eines Raums exponentiell von der Anzahl an Dimensionen abhängt. Insbesondere im Bereich der Modellierung realweltlicher Phänomene treten häufig hochdimensionale partielle Differentialgleichungen auf, die von dieser Problematik betroffen sind. Konkret bedingt der *curse of dimensionality* in diesen Fällen eine Laufzeit- und Speicherkomplexität, die exponentiell mit der Dimensionsanzahl wächst, wodurch klassische Lösungsverfahren ab einem gewissen Punkt unmöglich werden.
Vor diesem Hintergrund wird im ersten Teil dieser Arbeit mit dem hierarchischen Tuckerformat eine effiziente Tensorzerlegung eingeführt, die unter bestimmten Voraussetzungen keine exponentielle Abhängigkeit von der Anzahl an Dimensionen aufweist. Der zweite Teil widmet sich der Entwicklung eines epidemiologischen Modells, das als erweitertes SIR-Modell mit Diffusion unter Berücksichtigung von Altersklassen und Blutgruppen zusammengefasst werden kann. Der dritte Teil verbindet schließlich beide zuvor genannte Themen, indem ein spezielles Lösungsverfahren, das die Dynamik des Modells im hierarchischen Tuckerformat approximiert, ausgearbeitet wird. Im letzten Teil erfolgt schließlich die Berechnung mehrerer Simulationen in deren Diskussion der Fokus einerseits auf der Untersuchung der Leistungsfähigkeit des hierarchischen Tuckerformats und andererseits auf den epidemiologischen Eigenschaften des Modells liegt.
Die vorliegende Masterarbeit verfolgt somit das Ziel, nicht nur theoretische Grundlagen im Bereich des hierarchischen Tuckerformats zu präsentieren, sondern auch eine praxisnahe Anwendung im Kontext der epidemiologischen Modellierung zu bieten. Durch diese Synthese von Theorie und Anwendung wird versucht, einen Beitrag zu einem effizienten Umgang mit Daten aus hochdimensionalen Räumen zu leisten.

2 Mathematische Grundlage

Dieser erste Teil der Arbeit ist der Ausarbeitung des mathematischen Fundaments und der Etablierung einer einheitlichen Notation gewidmet. Hierbei vermittelt das erste Kapitel die grundlegenden Definitionen und Konzepte zu Tensoren. Darauf aufbauend erfolgt im zweiten Kapitel die Einführung des hierarchischen Tuckerformats, das eine effiziente Tensorzerlegung darstellt.

2.1 Tensoren

Tensoren sind vielseitige Objekte, zu denen es je nach behandelnder Disziplin unterschiedliche Zugänge gibt. Im Folgenden werden sie aus der Perspektive der Informatik als mehrdimensionale Arrays betrachtet, die als eine Erweiterung von Matrizen auf höhere Dimensionen zu verstehen sind.

2.1.1 Definition und Notation

Sei $A \in \mathbb{R}^{m \times n}$ eine Matrix über dem Körper der reellen Zahlen mit n-vielen Spalten und m-vielen Zeilen. Dann setzt sich A aus $m \cdot n$ vielen Einträgen $a_{ij} \in \mathbb{R}$ zusammen, wobei jedem Eintrag genau ein Indexpaar (i, j) mit $i \in \{1, ..., m\}$ und $j \in \{1, ..., n\}$ zugeordnet ist. Für die vorliegende Arbeit wird nun festgelegt, dass mit der Notation $A[i, j] = a_{ij}$ der Eintrag der i-ten Zeile und j-ten Spalte gemeint ist. Beschränkt man sich nicht nur auf Indexpaare, sondern erlaubt allgemeiner eine natürlich Zahl von Indizes zur Referenzierung von Einträgen, erhält man einen Tensor.

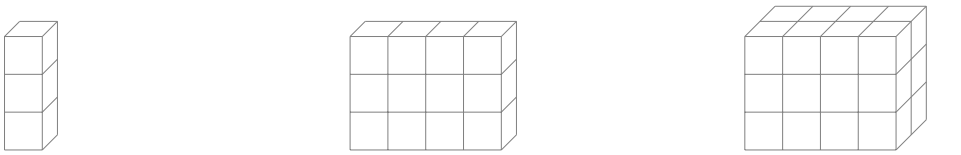
Definition 2.1 (Tensor)
Sei eine natürliche Zahl von Indexmengen $I_1, ..., I_d$ gegeben. Das kartesische Produkt dieser Indexmengen $\mathbf{I} = \times_{i=1}^{d} I_i$ enthält d-Tupel $i = (i_1, ..., i_d)$ mit $i_j \in I_j$ für alle $j \in \{1, ..., d\}$. Die Indizes i_j gehören zur j-ten *Dimension* beziehungsweise dem j-ten *Modus* des *Tensors*. Ein reellwertiger Tensor X mit Indexmenge $\mathbf{I}$ ist nun über seine Einträge

$$X_i = X[i] = X[i_1, ..., i_d] \in \mathbb{R}$$

definiert. Die entsprechende Menge der reellwertigen Tensoren mit Indexmenge $\mathbf{I}$ wird mit

$$\mathbb{R}^{\mathbf{I}} = \mathbb{R}^{I_1 \times ... \times I_d} := \{X = (X_i)_{i \in \mathbf{I}} \mid X_i = X[i] \in \mathbb{R} \text{ für alle } i \in \mathbf{I}\}$$

notiert. Die Länge d der Indextupel entspricht der Anzahl an Dimensionen beziehungsweise Modi des Tensors und wird *Ordnung* genannt.

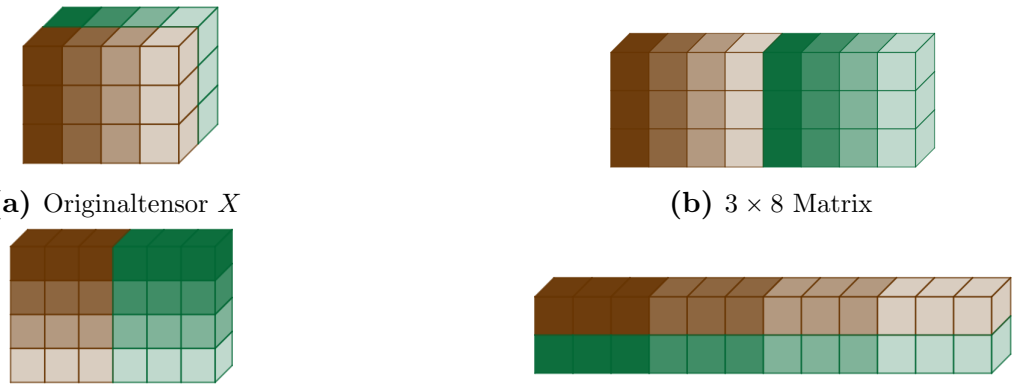


(a) 3 Tensor **(b)** 3×4 Tensor **(c)** $3 \times 4 \times 2$ Tensor

Abbildung 2.1: Darstellung von Tensoren erster, zweiter und dritter Ordnung.

Dieser Definition folgend entsprechen Vektoren Tensoren erster Ordnung und Matrizen Tensoren zweiter Ordnung. Zur kompakteren Notation der Tensorräume wird folgende Konvention festgelegt: $\mathbb{R}^{n_1 \times \ldots \times n_d} = \mathbb{R}^{\{1,\ldots,n_1\} \times \ldots \times \{1,\ldots,n_d\}}$. Des Weiteren seien mit $n_1 \times \ldots \times n_d$ Tensoren stets Tensoren aus dem $\mathbb{R}^{n_1 \times \ldots \times n_d}$ gemeint.

2.1.2 Matrizierung und Vektorisierung



(a) Originaltensor X **(b)** 3×8 Matrix

(c) 4×6 Matrix **(d)** 2×12 Matrix

Abbildung 2.2: Darstellung verschiedener Matrizierungen eines $3 \times 4 \times 2$ Tensors X.

Ausgehend von der Perspektive multidimensionaler Arrays liegt es nahe, die Anordnung der Einträge eines Tensors zu variieren. So ist es beispielsweise möglich, Tensoren beliebiger Ordnung als Matrizen oder Vektoren anzuordnen (siehe Abbildung 2.2). Es bedarf dazu einer neuen Indexmenge, die genau so viele Elemente wie die ursprüngliche Indexmenge enthält und einer Bijektion zwischen alter und neuer Indexmenge. Die gewählte Indexmenge prägt die Struktur der Modi des transformierten Tensors im Sinne von Anzahl und Größe. Die Bijektion bestimmt hingegen, nach welchem Schema die Tensoreinträge in die neue Struktur einzubringen sind.

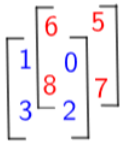


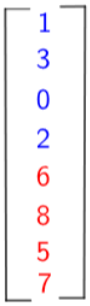
Abbildung 2.3: Tensor X aus Beispiel 2.2.

Beispiel 2.2
Sei $X \in \mathbb{R}^I$ mit $I = \{1,2\} \times \{1,2\} \times \{1,2\}$ ein reeller Tensor der Ordnung 3 dessen Einträge durch

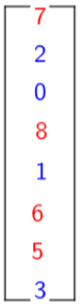
$$X[i,j,k] = 2i - j + 5(k-1)$$

definiert sind (siehe Abbildung 2.3). Entspricht die Indexmenge des transformierten Tensors beispielsweise $J = \{1,2,...,8\}$, so ist seine Struktur entsprechend Abbildung 2.4a gegeben. Zwischen den achtelementigen Indexmengen I und J existieren 8! verschiedene Bijektionen und damit 8! verschiedene Möglichkeiten, die vorgegebene Struktur mit Einträgen zu füllen. In diesem Beispiel werden die Bijektionen $\Phi : I \to J$ und $\Psi : I \to J$ gewählt, deren Definitionen Tabelle 2.1 zu entnehmen sind. Abbildungen 2.4b und 2.4c zeigen die zwei resultierenden transformierten Tensoren.

(a) Struktur des transformierten Tensors



(b) Transformierter Tensor gemäß Φ



(c) Transformierter Tensor gemäß Ψ

Abbildung 2.4: Transformierte Tensoren aus Beispiel 2.2.

(i,j,k)	$(1,1,1)$	$(2,1,1)$	$(1,2,1)$	$(2,2,1)$	$(1,1,2)$	$(2,1,2)$	$(1,2,2)$	$(2,2,2)$
$\Phi(i,j,k)$	1	2	3	4	5	6	7	8
$\Psi(i,j,k)$	5	8	3	2	6	4	7	1

Tabelle 2.1: Definition der Bijektionen Φ und Ψ aus Beispiel 2.2.

Im Allgemeinen existiert mehr als nur eine Bijektion zwischen zwei Indexmengen und prinzipiell funktionieren die im weiteren Verlauf behandelten Algorithmen, die mit derar-

tigen Tensortransformationen arbeiten, auch mit beliebigen Bijektionen, sofern diese konsequent beibehalten werden. Der Klarheit halber wird in dieser Arbeit trotzdem immer dieselbe Art von Bijektionen verwendet, die nun eingeführt wird.

Definition 2.3 (Kanonische Bijektion)
Für die Indexmenge $I = \{1, ..., n_1\} \times ... \times \{1, ..., n_d\}$ ist die *kanonische Bijektion* $\Phi_I : I \to \{1, ..., |I|\}$ wie folgt definiert:

$$\Phi_I(i_1, ..., i_d) = 1 + \sum_{j=1}^{d} (i_j - 1) \cdot \prod_{k=1}^{j-1} n_k$$

Die kanonische Bijektion Φ_I induziert eine Sortierung der Elemente von I in umgekehrt-lexikographischer Reihenfolge (siehe Anhang A.1). Folgendes Theorem stellt fest, dass es sich tatsächlich um eine Bijektion handelt.

Theorem 2.4 (Inverse der kanonischen Bijektion)
Die kanonische Bijektion Φ_I ist tatsächlich eine Bijektion und $\Phi_I^{-1} = (\phi_1(i), ..., \phi_d(i))$ mit

$$\phi_j(i) = \left\lfloor \frac{i - 1 - \sum_{k=j+1}^{d} (\phi_k(i) - 1) \prod_{l=1}^{k-1} n_l}{\prod_{k=1}^{j-1} n_k} \right\rfloor + 1$$
$$\text{und } \forall j \in \{1, ..., d\} : \ \phi_j(\Phi_I(i_1, ..., i_d)) = i_j$$

ist ihre inverse Funktion.

Beweis. Ein Beweis dieses Theorems findet sich unter Appendix A.1.1 in [2]. □

Unter Zuhilfenahme der kanonischen Bijektion formalisiert folgende Definition die Matrizierung eines Tensors.

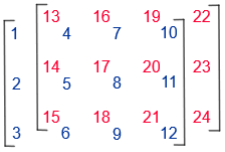
Definition 2.5 (Matrizierung)
Sei $X \in \mathbb{R}^{n_1 \times ... n_d}$ ein reeller Tensor der Ordnung d, $t = \{t_1, ..., t_k\}$ eine Teilmenge von $\{1, ..., d\}$ mit Komplement $[t] = \{[t]_1, ..., [t]_{d-k}\}$ und es gelte $\forall i \in \{1, ..., k-1\} : t_i < t_{i+1}$ sowie $\forall i \in \{1, ..., d-k-1\} : [t]_i < [t]_{i+1}$. Darauf aufbauend seien I_t und $I_{[t]}$ Indexmengen mit $I_t = \times_{i=1}^{k} \{1, ..., n_{t_i}\}$ und $I_{[t]} = \times_{i=1}^{d-k} \{1, ..., n_{[t]_i}\}$. Dann ist die Definition der t-Matrizierung von X, notiert als $\mathcal{M}_t(X)$, gegeben durch

$$\mathcal{M}_t(X) \in \mathbb{R}^{|I_t| \times |I_{[t]}|}$$
$$\mathcal{M}_t(X)[\Phi_{I_t}(i_{t_1}, ..., t_{t_k}), \Phi_{I_{[t]}}(i_{[t]_1}, ..., i_{[t]_{d-k}})] := X[i_1, ..., i_d].$$

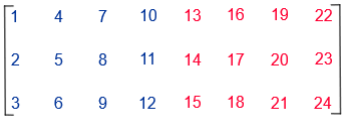
Hierbei sind Φ_{I_t} und $\Phi_{I_{[t]}}$ die jeweiligen kanonischen Bijektionen von I_t nach $\{1, ..., |I_t|\}$

beziehungsweise $I_{[t]}$ nach $\{1, ..., |I_{[t]}|\}$.

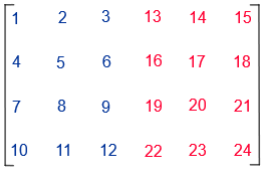
Für die Matrizierung $\mathcal{M}_t(X)$ eines Tensors $X \in \mathbb{R}^{n_1 \times ... \times n_d}$ gilt somit, dass die in t enthaltenen Modi von X in $\prod_{i \in t} n_i$ viele Zeilenindizes gruppiert werden und die übrigen Modi in $\prod_{i \in [t]} n_i$ viele Spaltenindizes. Durch die Wahl der kanonischen Bijektion geschieht dies in umgekehrt-lexikographischer Reihenfolge. Abbildung 2.5 illustriert dieses Vorgehen durch Darstellung verschiedener Matrizierungen desselben Tensors.



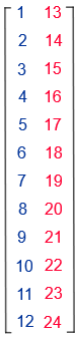
(a) Originaltensor X



(b) $\mathcal{M}_{\{1\}}(X) = \mathcal{M}_{\{2,3\}}(X)^T$



(c) $\mathcal{M}_{\{2\}}(X) = \mathcal{M}_{\{1,3\}}(X)^T$



(d) $\mathcal{M}_{\{3\}}(X)^T = \mathcal{M}_{\{1,2\}}(X)$

Abbildung 2.5: Abbildung verschiedener Matrizierungen eines $3 \times 4 \times 2$ Tensors X mit $X[i, j, k] = 1 + (i - 1) + (j - 1) \cdot 3 + (k - 1) \cdot 12$.

Fasst man alle Modi eines Tensors zu einem einzigen Modus zusammen, erhält man eine Vektorisierung des Tensors. Wie auch im Kontext der Matrizierung wird sich in dieser Arbeit auf eine einheitliche Art und Weise festgelegt, nach der Tensoren zu Vektoren transformiert werden.

Definition 2.6 (Vektorisierung)
Sei $X \in \mathbb{R}^I$ ein reeller Tensor von Ordnung d mit Indexmenge $I = \{1, ..., n_1\} \times ... \times \{1, ..., n_d\}$. Dann ist seine *Vektorisierung* $\text{vec}(X)$ durch

$$\text{vec}(X) \in \mathbb{R}^{|I|}$$
$$\text{vec}(X)[\Phi_I(i_1, ..., i_d)] := X[i_1, ..., i_d]$$

gegeben, wobei Φ_I der kanonischen Bijektion entspricht.

Begründet in der Isomorphie zwischen den beiden Vektorräumen $\mathbb{R}^{|I| \times 1}$ und $\mathbb{R}^{|I|}$ wird

$\text{vec}(X) \in \mathbb{R}^{|I|}$ fortan mit $\mathcal{M}_{\{1,...,d\}}(X) \in \mathbb{R}^{|I|\times 1}$ entsprechend der Vorschrift

$$\text{vec}(X)[i] = \mathcal{M}_{\{1,...,d\}}(X)[i, 1].$$

assoziiert.

2.1.3 Tensoroperationen

In diesem Abschnitt erfolgen Definition und Erläuterung grundlegender Rechenoperationen mit Tensoren.

2.1.3.1 Addition

Die Addition zweier Tensoren mit übereinstimmenden Dimensionen erfolgt genau wie die Addition zweier Matrizen elementweise.

Definition 2.7 (Addition)
Sei ein reeller Tensorraum $\mathbb{R}^{n_1\times...\times n_d}$ von beliebiger Ordnung $d \in \mathbb{N}$ gegeben. Dann ist die Addition zweier Tensoren aus diesem Raum wie folgt definiert:

$$\begin{aligned}
&+ : \mathbb{R}^{n_1\times...\times n_d} \times \mathbb{R}^{n_1\times...\times n_d} \to \mathbb{R}^{n_1\times...\times n_d} \\
&(X, Y) \mapsto X + Y \text{ mit} \\
&(X + Y)[i_1, ..., i_d] = X[i_1, ..., i_d] + Y[i_1, ..., i_d]
\end{aligned}$$

2.1.3.2 Skalarmultiplikation

Wie bei der Addition erfolgt die Multiplikation eines Tensors mit einem Skalar elementweise.

Definition 2.8 (Skalarmultiplikation)
Gegeben sei ein reeller Tensorraum $\mathbb{R}^{n_1\times...\times n_d}$ von beliebiger Ordnung $d \in \mathbb{N}$. Die Multiplikation von Tensoren aus gegebenem Raum mit einem reellen Skalar ist dann wie folgt definiert:

$$\begin{aligned}
&\cdot : \mathbb{R} \times \mathbb{R}^{n_1\times...\times n_d} \to \mathbb{R}^{n_1\times...\times n_d} \\
&(a, X) \mapsto a \cdot X \text{ mit} \\
&(a \cdot X)[i_1, ..., i_d] = a \cdot X[i_1, ..., i_d]
\end{aligned}$$

Proposition 2.9
Sei $\mathbf{I}$ eine Indexmenge. Dann ist $(\mathbb{R}^{\mathbf{I}}, +, \cdot)$ ein Vektorraum über $\mathbb{R}$.

Beweis. Da die Tensoraddition und die Multiplikation eines Tensors mit einem Skalar elementweise definiert sind, werden die notwendigen Eigenschaften zur Bildung eines Vektorraums direkt von der Addition beziehungsweise Multiplikation im Körper $\mathbb{R}$ geerbt. □

2.1.3.3 Permutation

Häufig ist es hilfreich, die Reihenfolge der Modi eines Tensors zu verändern. Gründe für eine Vertauschung können beispielsweise in einer einfacheren Notation oder konsistenten Darstellung der im Tensor kodierten Daten liegen. Um in der weiteren Arbeit Zugriff auf eine solche Tensorpermutation zu haben, führt die anschließende Definition eine entsprechende Operation ein.

Definition 2.10 (Tensorpermutation)
Sei $X \in \mathbb{R}^{n_1 \times \ldots \times n_d}$ ein reeller Tensor von Ordnung d und sei π eine d-stellige Permutation. Dann ist $per(\pi, X)$ der Tensor, der aus X hervorgeht, wenn die Modi von X entsprechend π vertauscht werden. $per(\pi, X)$ ist wie folgt definiert:

$$
\begin{aligned}
&per(\cdot,\cdot) : \mathbf{S}_d \times \mathbb{R}^{n_1 \times \ldots \times n_d} \to \mathbb{R}^{n_{\pi(1)} \times \ldots \times n_{\pi(d)}} \\
&(\pi, X) \mapsto per(\pi, X) \\
&per(\pi, X)[i_1, ..., i_d] := X[i_{\pi^{-1}(1)}, ..., i_{\pi^{-1}(d)}]
\end{aligned}
$$

$\mathbf{S}_d$ ist hierbei die symmetrische Gruppe mit Grad d.

Beispiel 2.11
Sei $M \in \mathbb{R}^{n \times m}$ eine reelle $n \times m$ Matrix beziehungsweise ein Tensor zweiter Ordnung und sei $\pi = (1\ 2)$ eine zweistellige Permutation. Dann entspricht $per(\pi, M) \in \mathbb{R}^{m \times n}$ der Transponierten M^T. Denn:

$$per(\pi, M)[i_1, i_2] = M[i_{\pi^{-1}(1)}, i_{\pi^{-1}(2)}] = M[i_2, i_1] = M^T[i_1, i_2]$$

(Hinweis: Die Permutation $\pi \in \mathbf{S}_2$ mit $\pi(1) = 2$ und $\pi(2) = 1$ wurde in diesem Beispiel in Zyklenschreibweise $(1\ 2)$ notiert.)

2.1.3.4 Indexkontraktion

Bei der Indexkontraktion werden zwei Tensoren mit mindestens teilweise übereinstimmenden Modi betrachtet. Die jeweiligen Tensoreinträge der zu kontrahierenden Modi werden dann analog zum inneren Produkt bei Vektoren multipliziert und aufaddiert. Die anschließende Definition setzt voraus, dass sich die zu kontrahierenden Modi beider Tensoren an jeweils letzter Stelle befinden. Dies stellt im Allgemeinen keine Einschränkung dar, da sich die Modi eines Tensors jederzeit umordnen lassen (siehe Def. 2.10).

Definition 2.12 (Indexkontraktion)
Seien $X \in \mathbb{R}^{n_1 \times ... \times n_d \times p_1 \times ... \times p_f}$ und $Y \in \mathbb{R}^{m_1 \times ... \times m_k \times p_1 \times ... \times p_f}$ zwei reelle Tensoren der Ordnung $d+f$ beziehungsweise $k+f$. Die *Indexkontraktion* der Modi $p_1, ..., p_f$ von X und Y ergibt einen Tensor der Ordnung $d+k$, der mit $\langle X, Y \rangle_{d+1,...,d+f}^{k+1,...,k+f}$ notiert wird und wie folgt definiert ist:

$$\langle X, Y \rangle_{d+1,...,d+f}^{k+1,...,k+f} \in \mathbb{R}^{n_1 \times ... \times n_d \times m_1 \times ... \times m_k}$$

$$\langle X, Y \rangle_{d+1,...,d+f}^{k+1,...,k+f}[i_1, ..., i_{d+k}] := \sum_{j_1=1}^{p_1} ... \sum_{j_f=1}^{p_f} X[i_1, ..., i_d, j_1, ..., j_f] \cdot Y[i_{d+1}, ..., i_{d+k}, j_1, ..., j_f]$$

Die beiden niedrig- beziehungsweise hochgestellten Zahlenfolgen $d+1, ..., d+f$ und $k+1, ..., k+f$ geben an, welche Modi kontrahiert werden. Die niedrigstehende Folge bezieht sich auf X und die hochstehende auf Y.

Die anschließenden Beispiele stellen klar, wie die Definition auf Fälle, bei denen die zu kontrahierenden Modi nicht an letzter Stelle stehen, verallgemeinert werden kann.

Beispiel 2.13
Sei A eine $n \times m$ und B eine $m \times l$ Matrix. Dann sind A und B gleichzeitig Tensoren zweiter Ordnung. Das Matrixprodukt $AB \in \mathbb{R}^{n \times l}$ lässt sich folgendermaßen als Indexkontraktion darstellen:

$$AB = \langle A, B \rangle_2^1, \quad \text{denn}$$
$$\langle A, B \rangle_2^1[i, j] = \sum_{k=1}^{m} A[i, k] \cdot B[k, j] = AB[i, j]$$

Beispiel 2.14
Seien $X \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ und $Y \in \mathbb{R}^{m_1 \times n_3 \times m_2 \times n_1 \times m_3}$. Dann ist die Indexkontraktion $\langle X, Y \rangle_{1,3}^{4,2} \in \mathbb{R}^{n_2 \times m_1 \times m_2 \times m_3}$ wie folgt über ihre Einträge bestimmt:

$$\langle X, Y \rangle_{1,3}^{4,2}[i_1, i_2, i_3, i_4] = \sum_{j_1=1}^{n_1} \sum_{j_2=1}^{n_3} X[j_1, i_1, j_2] \cdot Y[i_2, j_2, i_3, j_1, i_4]$$

Die Beispiele illustrieren den bereits genannten Aspekt dieser Notation, dass im Falle einer exemplarischen Indexkontraktion $\langle X, Y \rangle_{(a_i)_{i \in \{1,...,k\}}}^{(b_i)_{i \in \{1,...,k\}}}$ die Folgen $(a_i)_{i \in \{1,...,k\}}$ und $(b_i)_{i \in \{1,...,k\}}$ die Positionen der Indizes, über die summiert wird, anzeigen. Wenn also j_4 der vierte Summationsindex ist, so ist seine Position im Multiindex von X durch a_4 und von Y durch b_4 gegeben:

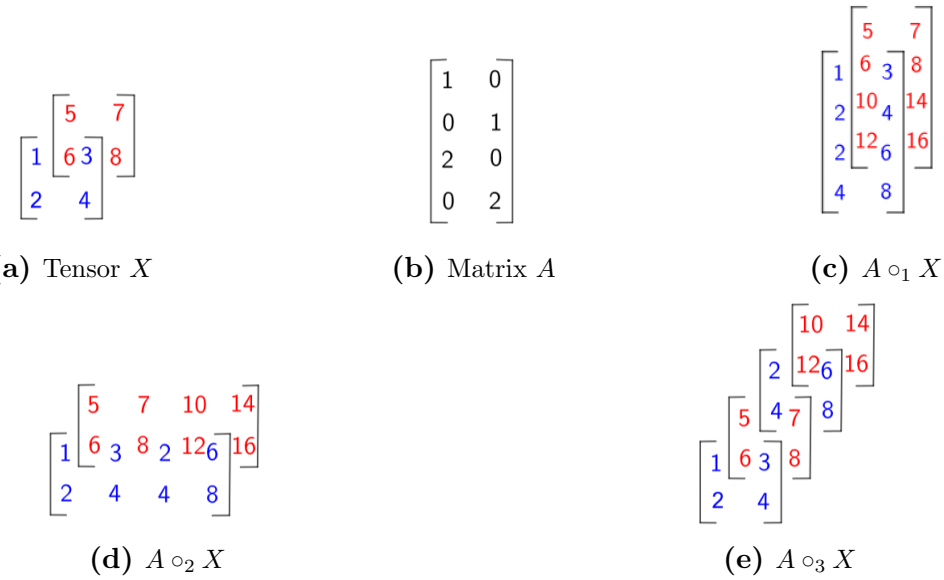
$$X[..., \underbrace{j_4}_{a_4\text{-te Stelle}}, ...] \text{ und } Y[..., \underbrace{j_4}_{b_4\text{-te Stelle}}, ...]$$

2.1.3.5 Modusmultiplikation

Die Multiplikation einer Matrix mit einem Tensor entlang eines definierten Modus wird als Modusmultiplikation (siehe Abbildung 2.6) bezeichnet und stellt einen Spezialfall einer Indexkontraktion dar.

Definition 2.15 (Modusmultiplikation)
Seien zwei reelle Tensorräume $\mathbb{R}^{n_1 \times \ldots \times n_d}$ und $\mathbb{R}^{m \times n_\mu}$ mit $\mu \in \{1, \ldots, d\}$ gegeben. Für $A \in \mathbb{R}^{m \times n_\mu}$ und $X \in \mathbb{R}^{n_1 \times \ldots \times n_d}$ bezeichnet der Ausdruck $A \circ_\mu X$ die *Modusmultiplikation* von A mit X entlang des Modus μ und ist wie folgt definiert:

$$\begin{aligned}
&\circ_\mu : \mathbb{R}^{m \times n_\mu} \times \mathbb{R}^{n_1 \times \ldots \times n_d} \to \mathbb{R}^{n_1 \times \ldots \times n_{\mu-1} \times m \times n_{\mu+1} \times \ldots \times n_d} \\
&(A, X) \mapsto A \circ_\mu X \\
&(A \circ_\mu X)[i_1, \ldots, i_d] := \sum_{j=1}^{n_\mu} A[i_\mu, j] X[i_1, \ldots, i_{\mu-1}, j, i_{\mu+1}, \ldots, i_d]
\end{aligned}$$



(a) Tensor X **(b)** Matrix A **(c)** $A \circ_1 X$

(d) $A \circ_2 X$ **(e)** $A \circ_3 X$

Abbildung 2.6: Modusmultiplikationen einer Matrix A mit einem Tensor X entlang der Modi 1, 2 und 3.

2.1.3.6 Multilineare Multiplikation

Gelegentlich tritt der Fall auf, dass jeder Modus eines Tensors mit einer Matrix multipliziert werden soll. Diese mehrfachen Modusmultiplikationen werden in folgender Definition als multilineare Multiplikation zusammengefasst.

Definition 2.16 (Multilineare Multiplikation)
Sei $X \in \mathbb{R}^{n_1 \times \ldots \times n_d}$ ein reeller Tensor der Ordnung d und seien $A_\mu \in \mathbb{R}^{m_\mu \times n_\mu}$ mit $\mu \in \{1, ..., d\}$ d viele Matrizen. Die *multilineare Multiplikation* der A_μ mit X, notiert als $(A_1, \ldots, A_d) \circ X$, ist wie folgt definiert:

$$(A_1, ..., A_d) \circ X := A_1 \circ_1 ... \; A_d \circ_d X \quad \in \mathbb{R}^{m_1 \times \ldots \times m_d}$$

Proposition 2.17
Die multilineare Multiplikation nach obiger Definition ist tatsächlich multilinear.

Beweis. Der Beweis findet sich im Anhang unter B.1. □

2.1.3.7 Elementweise Multiplikation

Die elementweise Multiplikation zweier Tensoren, die in ihren Modi übereinstimmen, ist, wie der Name bereits andeutet, durch das elementweise Multiplizieren der Tensoreinträge gegeben.

Definition 2.18 (Elementweise Multiplikation)
Sei ein reeller Tensorraum $\mathbb{R}^{n_1 \times \ldots \times n_d}$ von beliebiger Ordnung $d \in \mathbb{N}$ gegeben. Dann ist die *elementweise Multiplikation* zweier Tensoren aus diesem Raum wie folgt definiert:

$$\begin{aligned}
&\star : \mathbb{R}^{n_1 \times \ldots \times n_d} \times \mathbb{R}^{n_1 \times \ldots \times n_d} \to \mathbb{R}^{n_1 \times \ldots \times n_d} \\
&(X, Y) \mapsto X \star Y \\
&(X \star Y)[i_1, \ldots, i_d] := X[i_1, \ldots, i_d] \cdot Y[i_1, \ldots, i_d]
\end{aligned}$$

2.1.3.8 Äußeres Produkt

Vor der Einführung des Tensorprodukts soll zunächst die Definition des äußeren Produkts zweier Vektoren wiederholt werden.

Definition 2.19 (Äußeres Produkt von Vektoren)
Seien $v \in \mathbb{R}^n$ und $w \in \mathbb{R}^m$ zwei reelle Vektoren. Dann ist das *äußere Produkt* oder auch *dyadische Produkt* $v \otimes w$ definiert als:

$$\begin{aligned}
&\otimes : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^{n \times m} \\
&(v, w) \mapsto v \otimes w \\
&(v \otimes w)[i, j] := v[i] \cdot w[j]
\end{aligned}$$

Damit entspricht $v \otimes w$ einer Matrix mit n Zeilen und m Spalten.

Das äußere Produkt lässt sich geradewegs auf Tensoren verallgemeinern. In diesem Fall werden zwei Tensoren X und Y der Ordnung d beziehungsweise k zu einem Tensor $X \otimes Y$ der Ordnung $d + k$ verknüpft.

Definition 2.20 (Äußeres Produkt von Tensoren)
Seien zwei reelle Tensoren $X \in \mathbb{R}^{n_1 \times \ldots \times n_d}$ und $Y \in \mathbb{R}^{m_1 \times \ldots \times m_k}$ von Ordnung d beziehungsweise k gegeben. Das *äußere Produkt* oder auch *Tensorprodukt* $X \otimes Y$ ergibt einen reellen Tensor von Ordnung $d + k$, der wie folgt über seine Einträge definiert ist:

$$\begin{aligned}
&\otimes : \mathbb{R}^{n_1 \times \ldots \times n_d} \times \mathbb{R}^{m_1 \times \ldots \times m_k} \to \mathbb{R}^{n_1 \times \ldots \times n_d \times m_1 \times \ldots \times m_k} \\
&(X, Y) \mapsto X \otimes Y \\
&(X \otimes Y)[i_1, ..., i_{d+k}] := X[i_1, ..., i_d] \cdot Y[i_{d+1}, ..., i_{d+k}]
\end{aligned}$$

Aus der elementweisen Definition des äußeren Produkts geht hervor, dass es sich um eine bilineare Abbildung von $\mathbb{R}^{n_1 \times \ldots \times n_d} \times \mathbb{R}^{m_1 \times \ldots \times m_k}$ nach $\mathbb{R}^{n_1 \times \ldots \times n_d \times m_1 \times \ldots \times m_k}$ handelt.

Proposition 2.21
Das äußere Produkt zweier Tensoren ist eine bilineare Abbildung.

Beweis. Der Beweis findet sich im Anhang unter B.2. □

2.1.3.9 Elementweise Modusmultiplikation

Die elementweise Multiplikation zweier Tensoren wird nun durch die elementweise Multiplikation eines Vektors mit einem Tensor entlang eines vorgegebenen Modus ergänzt.

Definition 2.22 (Elementweise Modusmultiplikation)
Sei ein beliebiger reeller Tensorraum $\mathbb{R}^{n_1 \times \ldots \times n_d}$ von Ordnung $d \in \mathbb{N}$ gegegeben. Ferner sei $\mathbb{R}^{n_\mu}$ mit $\mu \in \{1, \ldots, d\}$ ein weiterer Tensorraum mit Ordnung 1. Die *elementweise Modusmultiplikation* von Tensoren beziehungsweise Vektoren aus $\mathbb{R}^{n_\mu}$ mit Tensoren aus $\mathbb{R}^{n_1 \times \ldots \times n_d}$ entlang des Modus μ ist dann definiert als:

$$\begin{aligned}
&\star_\mu : \mathbb{R}^{n_\mu} \times \mathbb{R}^{n_1 \times \ldots \times n_d} \to \mathbb{R}^{n_1 \times \ldots \times n_d} \\
&(v, X) \mapsto v \star_\mu X \\
&(v \star_\mu X)[i_1, \ldots, i_d] := v[i_\mu] \cdot X[i_1, \ldots, i_d]
\end{aligned}$$

Rechnung (2.1) zeigt, dass sich die elementweise Modusmultiplikation stets durch eine reguläre elementweise Multiplikation zweier Tensoren ausdrücken lässt. Seien also ein beliebiger Tensor $X \in \mathbb{R}^{n_1 \times \ldots \times n_d}$ und ein Vektor $v \in \mathbb{R}^{n_\mu}$ mit $\mu \in \{1, \ldots, d\}$ gegeben. Dann ergibt

sich die elementweise Modusmultiplikation von v mit X entlang des Modus μ als

$$\begin{aligned}
&\left((\mathbf{1}_{n1} \otimes \ldots \otimes \mathbf{1}_{n_{\mu-1}} \otimes v \otimes \mathbf{1}_{n_{\mu+1}} \otimes \ldots \otimes \mathbf{1}_{n_d}) \star X\right)[i_1, \ldots, i_d] \\
&= (\mathbf{1}_{n1} \otimes \ldots \otimes \mathbf{1}_{n_{\mu-1}} \otimes v \otimes \mathbf{1}_{n_{\mu+1}} \otimes \ldots \otimes \mathbf{1}_{n_d})[i_1, \ldots, i_d] \cdot X[i_1, \ldots, i_d] \\
&= \mathbf{1}_{n_1}[i_1] \cdot \ldots \cdot \mathbf{1}_{n_{\mu-1}}[i_{\mu-1}] \cdot v[i_\mu] \cdot \mathbf{1}_{n_{\mu+1}}[i_{\mu+1}] \cdot \ldots \cdot \mathbf{1}_{n_d}[i_d] \cdot X[i_1, \ldots, i_d] \\
&= v[i_\mu] \cdot X[i_1, \ldots, i_d] \\
&= (v \star_\mu X)[i_1, \ldots, i_d],
\end{aligned} \tag{2.1}$$

wobei die $\mathbf{1}_{n_i}$ Vektoren bestehend aus jeweils n_i Einsen sind: $\mathbf{1}_{n_i} := (1, \ldots, 1) \in \mathbb{R}^{n_i}$.

2.1.3.10 Norm

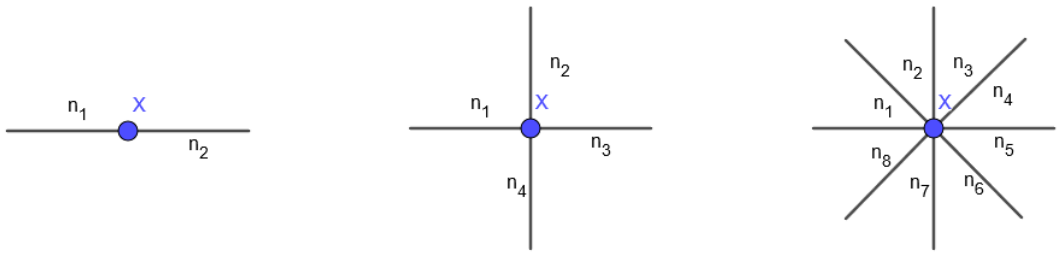
Die Frobeniusnorm einer reellen Matrix entspricht der Quadratwurzel der Summe aller quadrierten Matrixelemente. Analog wird im Folgenden die Frobeniusnorm für reelle Tensoren definiert.

Definition 2.23 (Frobeniusnorm)
Sei $X \in \mathbb{R}^{n_1 \times \ldots \times n_d}$ ein reeller Tensor von Ordnung d. Dann ist die zugehörige *Frobeniusnorm* $||X||_F$ wie folgt definiert:

$$||X||_F := \sqrt{\sum_{i_1=1}^{n_1} \ldots \sum_{i_d=1}^{n_d} X[i_1, \ldots, i_d]^2}$$

Abkürzend wird in der restlichen Arbeit stets $||X||$ anstelle von $||X||_F$ geschrieben.

2.1.4 Graphische Notation



(a) $n_1 \times n_2$ Tensor **(b)** $n_1 \times n_2 \times n_3 \times n_4$ Tensor **(c)** $n_1 \times \ldots \times n_8$ Tensor

Abbildung 2.7: Graphische Darstellung von Tensoren unterschiedlicher Ordnung.

Oft tragen Diagramme wesentlich dazu bei, Tensoroperationen verständlich zu machen. Daher wird nun eine [2] entnommene graphische Darstellung von Tensoren eingeführt, die es ermöglicht, Indexkontraktionen zu visualisieren. Dem Paradigma dieser Notation folgend,

werden Tensoren $X \in \mathbb{R}^{n_1 \times \ldots \times n_d}$ als Kreise und ihre d Modi als Äste, die ihren Ursprung am Kreisrand haben, repräsentiert (siehe Abbildung 2.7). Indexkontraktionen werden abgebildet, indem sich die Kreise der verrechneten Tensoren die Äste der zu kontrahierenden Modi teilen. Die nachstehenden Beispiele illustrieren verschiedenste Konstellationen.

Beispiel 2.24 (Matrix-Vektor-Multiplikation)
Sei A eine $n_1 \times n_2$ Matrix und B ein Vektor mit n_2 Einträgen. Die Indexkontraktion von A mit B via n_2 entspricht der üblichen Matrix-Vektor-Multiplikation und ist gegeben als:

$$\langle A, B \rangle_2^1[i] = \sum_{j=1}^{n_2} A[i,j] \cdot B[j]$$

Die diagrammatische Darstellung stellt sich wie folgt dar:

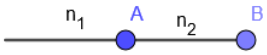


Abbildung 2.8: Matrix-Vektor-Multiplikation

Beispiel 2.25 (Matrixmultiplikation)
Sei A eine $n_1 \times n_2$ und B eine $n_2 \times n_3$ Matrix. Die Formulierung des Matrixprodukts AB als Indexkontraktion ist durch

$$\langle A, B \rangle_2^1[i,j] = \sum_{k=1}^{n_2} A[i,k] \cdot B[k,j]$$

gegeben. Das entsprechende Diagramm ergibt sich wiederum als:

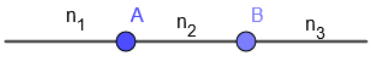


Abbildung 2.9: Matrixmultiplikation

Beispiel 2.26 (Modusmultiplikation)
Sei A eine $n_1 \times n_2$ Matrix und B ein $n_2 \times n_3 \times n_4$ Tensor. Die Multiplikation von A mit B entlang dessen ersten Modus lautet:

$$(A \circ_1 B)[i,j,k] = \sum_{l=1}^{n_2} A[i,l] \cdot B[l,j,k]$$

Die zugehörige graphische Repräsentation stellt sich wie folgt dar:

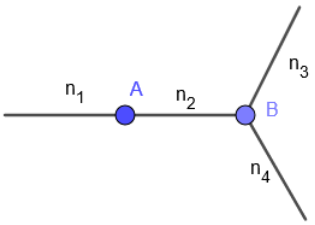


Abbildung 2.10: Modusmultiplikation

Beispiel 2.27 (Indexkontraktion)
Sei A ein $n_1 \times n_2 \times n_3 \times n_4$ Tensor und B ein $n_2 \times n_5 \times n_3$ Tensor. Die Indexkontraktion von A mit B über die Modi $2, 3$ beziehungsweise $1, 3$ folgt nachstehender Formel:

$$\langle A, B \rangle_{2,3}^{1,3}[i, j, k] = \sum_{l=1}^{n_2} \sum_{m=1}^{n_3} A[i, l, m, j] \cdot B[l, k, m]$$

Das zugehörige Diagramm sieht folgendermaßen aus:

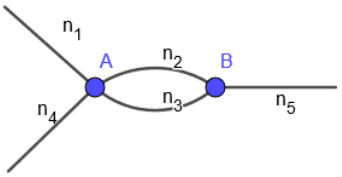


Abbildung 2.11: Indexkontraktion

Beispiel 2.28 (Tensornetzwerk)
Sei A ein $n_1 \times n_2 \times n_3$ Tensor, B ein $n_2 \times n_4 \times n_5 \times n_6$ Tensor, C ein $n_5 \times n_7 \times n_6$ Tensor und D ein $n_7 \times n_8$ Tensor. Dann lassen sich die verschachtelten Indexkontraktionen

$$\langle \langle \langle A, B \rangle_2^1, C \rangle_{4,5}^{1,3}, D \rangle_4^1$$

mit

$$\begin{aligned} &\langle A, B \rangle_2^1 \in \mathbb{R}^{n_1 \times n_3 \times n_4 \times n_5 \times n_6} \quad \text{und} \\ &\langle \langle A, B \rangle_2^1, C \rangle_{4,5}^{1,3} \in \mathbb{R}^{n_1 \times n_3 \times n_4 \times n_7} \quad \text{sowie} \\ &\langle \langle \langle A, B \rangle_2^1, C \rangle_{4,5}^{1,3}, D \rangle_4^1 \in \mathbb{R}^{n_1 \times n_3 \times n_4 \times n_8} \end{aligned}$$

auch deutlich übersichtlicher als Diagramm darstellen:

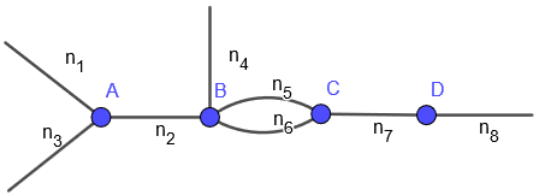


Abbildung 2.12: Tensornetzwerk

Der nach Durchführung aller Indexkontraktionen resultierende Tensor hat schließlich folgende Gestalt:

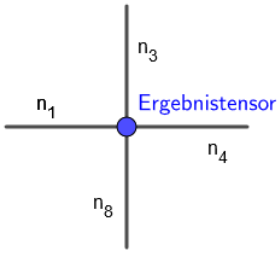


Abbildung 2.13: Kontrahiertes Tensornetzwerk aus Abbildung 2.12

2.1.5 Speicherkomplexität

Ein Tensor $X \in \mathbb{R}^{n_1 \times \ldots \times n_d}$ hat gemäß seiner d Modi $\prod_{\mu=1}^{d} n_\mu$ viele Einträge. Damit kann seine Speicherkomplexität durch

$$\text{Speicher}(X) \leqslant n^d \quad n = \max_{\mu}(n_\mu)$$

abgeschätzt werden. Anhand dieser Abschätzung ist direkt ersichtlich, dass die Speicherkomplexität mit jeder Dimension exponentiell ansteigt. Dies führt zu einer Reihe von Schwierigkeiten, die dem durch den Mathematiker Richard Bellman benannten Phänomen des *curse of dimensionality* zuzuordnen sind.

2.2 Hierarchisches Tuckerformat

Dieses Kapitel ist der Einführung des hierarchischen Tuckerformats gewidmet. Das hierarchische Tuckerformat von Tensoren ist eine Spezialisierung des allgemeinen Tuckerformats und bietet genau wie dieses die Möglichkeit, Tensoren auf der Grundlage von Singulärwertzerlegungen zu approximieren. Unter günstigen Umständen kann damit die exponentielle Abhängigkeit von der Anzahl an Dimensionen aufgehoben werden. Die Entwicklung des hierarchischen Tuckerformats geht auf Hackbusch und Kühn zurück, die zu ihm

erstmals 2009 publiziert haben [4]. Die Ausführungen dieses Kapitels basieren sich jedoch auf einer Arbeit von Grasedyck [3] und einer von Kressner und Tobler [11]. Im nächsten Abschnitt erfolgt zuerst die Skizzierung des allgemeinen Tuckerformats, um im Anschluss darauf aufbauend, das hierarchische Tuckerformat im Detail erläutern zu können.

2.2.1 Klassisches Tuckerformat

Die Idee des Tuckerformats liegt darin, einen Tensor in einen Kerntensor und eine Menge von Matrizen zu zerlegen (siehe Abbildung 2.14). Der Originaltensor kann dabei durch Modusmultiplikationen dieser Matrizen mit dem Kerntensor wiederhergestellt werden (vgl. Abbildung 2.15). Während Ledyard Tucker Namensstifter des Tuckerformats [22] ist, findet sich sein Ursprung in einer Arbeit von Frank Hitchcock [7].

Definition 2.29 (Tuckerrang)
Sei $X \in \mathbb{R}^{n_1 \times ... \times n_d}$ ein reeller Tensor der Ordnung d. Mit dem *Tuckerrang* von X wird das elementweise minimale Tupel $(r_1, ..., r_d)$ bezeichnet, für das spaltenweise orthonormale Matrizen $U_\mu \in \mathbb{R}^{n_\mu \times r_\mu}$ mit $\mu \in \{1, ..., d\}$ und ein Tensor $G \in \mathbb{R}^{r_1 \times ... \times r_d}$ existieren, die die folgende Gleichung erfüllen

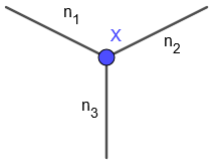
$$X = (U_1, ..., U_d) \circ G.$$

Eine derartige Darstellung trägt den Namen *orthogonales Tuckerformat* beziehungsweise wird $X = (U_1, ..., U_d) \circ G$ direkt *orthogonaler Tuckertensor* genannt. Sind die Matrizen U_μ nicht spaltenweise orthonormal, wird vom *Tuckerformat* beziehungsweise *Tuckertensor* gesprochen. Die Menge aller Tuckertensoren zu einem gegebenen Tuckerrang $(r_1, ..., r_d)$ wird mit $Tucker(r_1, ..., r_d)$ notiert. Des Weiteren werden die Matrizen $U_1, ..., U_d$ des (orthogonalen) Tuckerformats als *Modusmatrizen* bezeichnet.

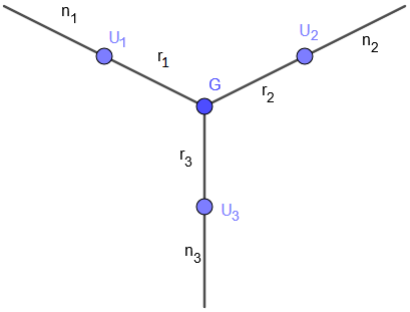
Der Speicherbedarf eines Tuckertensors $X = (U_1, ..., U_d) \circ G$ kann durch

$$\text{Speicher}((U_1, ..., U_d), G) \leqslant d \cdot n \cdot r + r^d \quad n = \max_\mu(n_\mu),\ r = \max_\mu(r_\mu)$$

abgeschätzt werden und skaliert damit exponentiell mit der Anzahl an Modi d des zugrundeliegenden Tensors. Damit ist die Behandlung hochdimensionaler Tensoren auch im Tuckerformat nur äußerst schwierig umsetzbar. Trotzdem wird deutlich, dass im Vergleich zum vollen Tensor umso mehr Speicherplatz eingespart werden kann, je kleiner der Tuckerrang $(r_1, ..., r_d)$ im Vergleich zu den ursprünglichen Modusgrößen $n_1, ..., n_d$ ist.
Sind Modusmatrizen mit orthonormalen Spalten $(U_1, \ldots, U_d)$ zu einem Tensor X von Ordnung d gegeben, minimiert der eindeutige Kerntensor $G = (U_1^T, ..., U_d^T) \circ X$ die Gleichung $||X - (U_1, ..., U_d) \circ G||$. Auf dieser Einsicht fußt folgende Operation, die einen vollen Tensor ins orthogonale Tuckerformat überführt.



(a) X als normaler Tensor



(b) X im Tuckerformat mit $X = (U_1, U_2, U_3) \circ G$

Abbildung 2.14: Diagrammatische Darstellung eines Tensors X von Ordnung drei.

Definition 2.30 (Tuckerkürzung)
Sei $X \in \mathbb{R}^{n_1 \times \ldots \times n_d}$ ein reeller Tensor und sei $U_\mu \cdot \Sigma_\mu \cdot V_\mu^T$ eine Singulärwertzerlegung von $\mathcal{M}_\mu(X)$ mit $U_\mu \in \mathbb{R}^{n_\mu \times n_\mu}$ und $\Sigma_\mu = \text{diag}(\sigma_{\mu,1}, ..., \sigma_{\mu,n_\mu})$. Der auf den Tuckerrang $(r_1, ..., r_d)$ gekürzte orthogonale Tuckertensor von X ist dann durch

$$\mathcal{T}_{(r_1,...,r_d)}(X) = (\tilde{U}_1 \cdot \tilde{U}_1^T, ..., \tilde{U}_d \cdot \tilde{U}_d^T) \circ X = (\tilde{U}_1, ..., \tilde{U}_d) \circ \left((\tilde{U}_1^T, ..., \tilde{U}_d^T) \circ X \right)$$

gegeben, wobei die Matrizen $\tilde{U}_\mu$ aus den r_μ dominanten Singulärvektoren in U_μ bestehen.

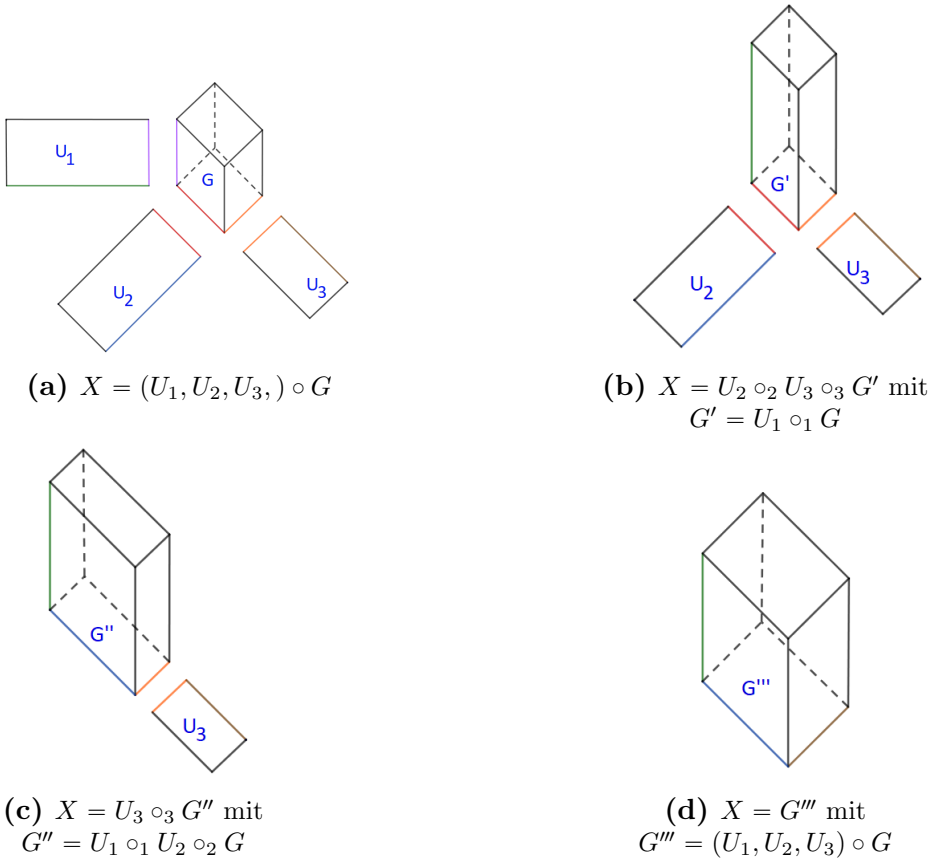
Durch den frei wählbaren Tuckerrang bei der Tuckerkürzung besteht die Möglichkeit, einen vollen Tensor ins Tuckerformat zu überführen und dabei zu approximieren. Werden keine Singulärvektoren weggelassen und der volle Tensor exakt im Tuckerformat abgebildet, spricht man von einer Singulärwertzerlegung höherer Ordnung. Ansonsten ist der resultierende Kürzungsfehler, durch die zu den weggelassenen Singulärvektoren gehörigen Singulärwerte abzuschätzen.

Lemma 2.31 (Tuckerapproximation)
Sei $X \in \mathbb{R}^{n_1 \times \ldots \times n_d}$ ein reeller Tensor von Ordnung d. Dann ist der Kürzungsfehler der Tuckerkürzung von X mit Tuckerrang $(r_1, ..., r_d)$ durch

$$||X - \mathcal{T}_{(r_1,...,r_d)}(X)|| \leqslant \sqrt{\sum_{i=1}^{d} \sum_{j=r_i+1}^{n_i} \sigma_{i,j}^2}$$

nach oben beschränkt.

Beweis. Das Lemma entspricht Property 10 in [13]. □



(a) $X = (U_1, U_2, U_3,) \circ G$

(b) $X = U_2 \circ_2 U_3 \circ_3 G'$ mit $G' = U_1 \circ_1 G$

(c) $X = U_3 \circ_3 G''$ mit $G'' = U_1 \circ_1 U_2 \circ_2 G$

(d) $X = G'''$ mit $G''' = (U_1, U_2, U_3) \circ G$

Abbildung 2.15: Wiederherstellung eines Originaltensors X aus gegebenem Tuckertensor $X = (U_1, U_2, U_3) \circ G$.

2.2.2 Hierarchisches Tuckerformat: Definition und Notation

Das hierarchische Tuckerformat ist eine mehrstufige Variante des Tuckerformats. Während das Tuckerformat zu jedem Modus des vollen Tensors eine separate Matrix verfügt, werden beim hierarchischen Tuckerformat die Modi des vollen Tensors als Baum angeordnet und anschließend zu jedem Knoten, der wiederum einer Menge von Modi entspricht, eine separate Matrix angelegt.

Definition 2.32 (Dimensionsbaum)
Ein *Dimensionsbaum* T_I zu einer Indexmenge $I = \{1, ..., n_1\} \times ... \times \{1, ..., n_d\}$ ist ein Binärbaum mit Wurzel $\text{Root}(T_I) = \{1, ..., d\}$ und Tiefe $p = \lceil \log_2(d) \rceil$, dessen Knoten als *Modus-* oder *Dimensionscluster* bezeichnet werden. Jeder Moduscluster $t \in T_I$ ist entweder

1. ein Blattknoten und einelementige Menge $t = \{\mu\}$ $(\mu \in \{1, ..., d\})$ mit einer Distanz von p beziehungsweise $p - 1$ zur Wurzel oder

2. ein innerer Knoten und damit die Vereinigungsmenge $t = s_1 \dot\cup s_2$ seiner zwei direkten Nachfahren $\{s_1, s_2\} = S(t)$.

Das *Level* l von T_I wird als die Menge

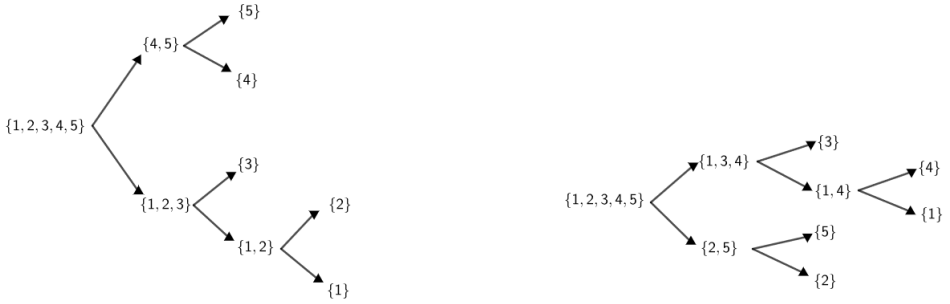
$$T_I^l := \{t \in T_I \mid t \text{ hat eine Distanz von } l \text{ zum Wurzelknoten Root}(T_I)\}$$

definiert. Ferner wird die Menge aller Blattknoten mit $\mathcal{L}(T_I)$ und die Menge aller inneren Knoten mit $\mathcal{I}(T_I)$ notiert. Entsprechend gilt: $T_I = \mathcal{L}(T_I) \dot{\cup} \mathcal{I}(T_I)$.

Diese Definition lässt offen, wie die Elemente eines Knotens auf seine direkten Nachfahren aufgeteilt werden. Im weiteren Verlauf dieser Arbeit wird stets von kanonischen Dimensionsbäumen ausgegangen, die folgende Eigenschaft erfüllen: Ist $t = \{\mu_1, ..., \mu_m\}$ innerer Knoten eines kanonischen Dimensionsbaums, so gilt für seine beiden Nachfolgeknoten

$$t_1 := \{\mu_1, ..., \mu_r\},\ t_2 := \{\mu_{r+1}, ..., \mu_m\} \text{ mit } r = \left\lceil \frac{m}{2} \right\rceil.$$

Abbildung 2.16 veranschaulicht den Unterschied zwischen kanonischen und nicht kanonischen Dimensionsbäumen.



(a) Kanonischer Dimensionsbaum **(b)** Nicht-kanonischer Dimensionsbaum

Abbildung 2.16: Abbildung zweier Dimensionsbäume mit $d = 5$.

Lemma 2.33
Die Knoten auf jedem Level des kanonischen Dimensionsbaums T_I mit Tiefe p bilden disjunkte Teilmengen von $\{1, ..., d\}$. Die Anzahl der Knoten auf Level l ist durch

$$\#T_I^l = \begin{cases} 2^l & l < p \\ 2 \cdot d - 2^p & l = p \end{cases}$$

gegeben. Ferner gilt $\#T_I = 2 \cdot d - 1$ mit $\#\mathcal{I}(T_I) = d - 1$ und $\#\mathcal{L}(T_I) = d$.

Beweis. Das Lemma entspricht Lemma 8 in [3]. □

Die durch den kanonischen Dimensionsbaum gestaltete Hierarchie zwischen den verschiedenen Modi eines Tensors wird nun genutzt, um den hierarchischen Rang eines Tensors zu definieren.

Definition 2.34 (Hierarchischer Rang)
Sei $X \in \mathbb{R}^I = \mathbb{R}^{n_1 \times \ldots \times n_d}$ ein reeller Tensor der Ordnung d und T_I der zugehörige kanonische Dimensionsbaum. Der *hierarchische Rang* von X ist eine Familie $(k_t)_{t \in T_I}$, für die gilt:

$$\forall t \in T_I : k_t = \operatorname{rank}(\mathcal{M}_t(X))$$

Die Menge aller Tensoren deren hierarchischer Rang knotenweise höchstens $(k_t)_{t \in T_I}$ entspricht, wird als

$$\mathcal{H}\text{-Tucker}((k_t)_{t \in T_I}) := \left\{ X \in \mathbb{R}^I \mid \forall t \in T_I : \operatorname{rank}(\mathcal{M}_t(X)) \leqslant k_t \right\}$$

notiert.

Anmerkung 2.35
Für den Rang eines Wurzelknotens gilt wegen $\mathcal{M}_{\mathrm{Root}(T_I)}(X) \in \mathbb{R}^{(\prod_{\mu \in T_I} n_\mu) \times 1}$ stets $k_{\mathrm{Root}(T_I)} = \operatorname{rank}\left(\mathcal{M}_{\mathrm{Root}(T_I)}(X)\right) = 1$.

Zentral für die Konstruktion eines hierarchischen Tuckertensors aus gegebenem vollen Tensor ist folgendes Lemma, das die Verschachtelung der Spaltenräume der Matrizierungen des vollen Tensors behandelt.

Lemma 2.36 (Verschachtelungslemma)
Sei $X \in \mathbb{R}^{n_1 \times \ldots \times n_d}$ ein Tensor von Ordnung d, T_I der zugehörige kanonische Dimensionsbaum und $(k_t)_{t \in T_I}$ der hierarchische Rang von X. Ferner sei $U_t \in \mathbb{R}^{\prod_{\mu \in t} n_\mu \times k_t}$ mit $t \in T_I$ eine Basis für den Spaltenraum der t-Matrizierung $\mathcal{M}_t(X)$. Dann gilt für jeden Moduscluster $t \in T_I$ mit direkten Nachfahren $S(t) = \{t_1, t_2\}$

$$\operatorname{image}(U_t) \subseteq \operatorname{image}\left(U_{t_2} \otimes_{\mathcal{K}} U_{t_1}\right), \tag{2.2}$$

wobei mit $\otimes_{\mathcal{K}}$ das Kronecker-Produkt (siehe Anhang A.2) angezeigt wird und $U_{t_1} \in \mathbb{R}^{\prod_{\mu \in t_1} n_\mu \times k_{t_1}}$ sowie $U_{t_2} \in \mathbb{R}^{\prod_{\mu \in t_2} n_\mu \times k_{t_2}}$ Basen für die Spaltenräume der Matrizierungen $\mathcal{M}_{t_1}(X)$ beziehungsweise $\mathcal{M}_{t_2}(X)$ sind.

Beweis. Das Lemma entspricht dem Lemma 2.1 in [11]. □

Aus (2.2) des obigen Lemmas folgt die Existenz sogenannter *Transfertensoren* $B_t \in \mathbb{R}^{k_{t_1} \times k_{t_2} \times k_t}$ mit

$$U_t = (U_{t_2} \otimes_{\mathcal{K}} U_{t_1}) \mathcal{M}_{\{1,2\}}(B_t) \tag{2.3}$$

bzw. äquivalent

$$U_t = \mathcal{M}_{\{1,2\}}(U_{t_1} \circ_1 U_{t_2} \circ_2 B_t). \tag{2.4}$$

Damit genügt es für alle inneren Knoten $t \in \mathcal{I}(T_I)$ die Transfertensoren $B_t \in \mathbb{R}^{k_{t_1} \times k_{t_2} \times k_t}$ abzuspeichern, da die $U_t \in \mathbb{R}^{\prod_{\mu \in t} n_\mu \times k_t}$ durch rekursive Anwendung von (2.3) beziehungsweise (2.4) gebildet werden können. Lediglich für die Blattknoten $t = \{\mu\} \in \mathcal{L}(T_I)$ müssen die $U_t \in \mathbb{R}^{n_\mu \times k_t}$ vorgehalten werden. Häufig ist es nützlich, die Transfertensoren in eine matrizierte Form $\mathcal{M}_{1,2}(B_t) \in \mathbb{R}^{k_{t_1} \cdot k_{t_2} \times k_t}$ zu überführen. Da dies eine umkehrbare Aktion ist und in der Regel aus dem Kontext ersichtlich ist, in welcher Form ein Transfertensor vorliegt, wird im weiteren Verlauf stets B_t geschrieben und die Matrizierung $\mathcal{M}_{\{1,2\}}(B_t)$ nur in Ausnahmefällen explizit angezeigt. Außerdem wird bei der Notation von tief- oder hochgestellten Modusclustern $t \in T_I$ gegebenenfalls auf Mengenklammern und Kommata verzichtet, wodurch sich folgende Darstellung ergibt: U_{12} statt $U_{\{1,2\}}$, $\mathcal{M}_{34}$ anstelle von $\mathcal{M}_{\{3,4\}}$ und so weiter.

Beispiel 2.37
Wiederholte Anwendung von (2.4) auf einen Tensor X von Ordnung vier ergibt:

$$\begin{aligned}\operatorname{vec}(X) = \mathcal{M}_{1234}(X) &= \mathcal{M}_{12}(U_{12} \circ_1 U_{34} \circ_2 B_{1234}) \\ &= \mathcal{M}_{12}\left(\underbrace{\mathcal{M}_{12}(U_1 \circ_1 U_2 \circ_2 B_{12})}_{=U_{12}} \circ_1 \underbrace{\mathcal{M}_{12}(U_3 \circ_1 U_4 \circ_2 B_{34})}_{=U_{34}} \circ_2 B_{1234}\right)\end{aligned}$$

Äquivalent erhält man mit (2.3):

$$\begin{aligned}\operatorname{vec}(X) = \mathcal{M}_{1234}(X) &= (U_{34} \otimes_{\mathcal{K}} U_{12}) B_{1234} \\ &= \left(\underbrace{(U_4 \otimes_{\mathcal{K}} U_3) B_{34}}_{=U_{34}} \otimes_{\mathcal{K}} \underbrace{(U_2 \otimes_{\mathcal{K}} U_1) B_{12}}_{=U_{12}}\right) B_{1234} \\ &= (U_4 \otimes_{\mathcal{K}} U_3 \otimes_{\mathcal{K}} U_2 \otimes_{\mathcal{K}} U_1)(B_{34} \otimes_{\mathcal{K}} B_{12}) B_{1234}\end{aligned}$$

Definition 2.38 (Hierarchisches Tuckerformat)
Sei I eine Indexmenge und T_I der zugehörige kanonische Dimensionsbaum. Des Weiteren sei $(k_t)_{t \in T_I}$ eine Rangverteilung auf T_I, $X \in \mathcal{H}\text{-Tucker}((k_t)_{t \in T_I})$ und $(U_t)_{t \in T_I}$ eine Familie von Matrizen mit

$$\forall t \in T_I : U_t \text{ ist Basis von image}(\mathcal{M}_t(X)).$$

Gemeinsam mit der Familie der Transfertensoren $(B_t)_{t \in \mathcal{I}(\mathcal{T}_\mathcal{I})}$, die bezogen auf $(U_t)_{t \in T_I}$ (2.4) beziehungsweise (2.3) erfüllen, ist das Tupel

$$((U_t)_{t \in \mathcal{L}(T_I)}, (B_t)_{t \in \mathcal{I}(\mathcal{T}_\mathcal{I})}) \tag{2.5}$$

eine Repräsentation von X im *hierarchischen Tuckerformat*, die auch als *hierarchischer Tuckertensor* bezeichnet wird. Sind die Spalten aller Matrizen $(U_t)_{t\in T_I}$ orthonormal, wird $((U_t)_{t\in\mathcal{L}(T_I)}, (B_t)_{t\in\mathcal{I}(\mathcal{T}_\mathcal{I})})$ *orthogonales hierarchisches Tuckerformat* beziehungsweise *orthogonaler hierarchischer Tuckertensor* genannt.

Lemma 2.39 (Speicherkomplexität)
Sei $I = \{1, ..., n_1\} \times ... \times \{1, ..., n_d\}$ eine Indexmenge und $X \in \mathcal{H}\text{-Tucker}((k_t)_{t\in T_I})$ mit hierarchischem Tuckerformat $((U_t)_{t\in\mathcal{L}(T_I)}, (B_t)_{t\in\mathcal{I}(T_I)})$. Dann ist der Speicherbedarf des hierarchischem Tuckerformats von X durch

$$\begin{aligned} &\text{Speicher}(((U_t)_{t\in\mathcal{L}(T_I)}, (B_t)_{t\in\mathcal{I}(T_I)})) \leqslant (d-2)k^3 + k^2 + dnk \\ &\text{mit } n = \max_{\mu\in\{1,...,d\}}(n_\mu) \text{ und } k = \max_{t\in T_I}(k_t) \end{aligned} \tag{2.6}$$

beschränkt.

Beweis. Für jeden inneren Knoten $t \in \mathcal{I}(T_I)$ mit direkten Nachfahren $S(t) = \{t_1, t_2\}$ liegt genau ein Transfertensor $B_t \in \mathbb{R}^{k_{t_1}\times k_{t_2}\times k_t}$ vor. Da $k_{\text{Root}(T_I)} = 1$ immer gültig ist und T_I genau $d-2$ innere Knoten besitzt, die nicht die Wurzel sind, erhalten wir:

$$\text{Speicher}((B_t)_{t\in\mathcal{I}(T_I)}) \leqslant (d-2)k^3 + k^2 \qquad k = \max_{t\in\mathcal{I}(T_I)} \tag{2.7}$$

Für jeden der d-vielen Blattknoten $t = \{\mu\} \in \mathcal{L}(T_I)$ existiert genau eine Matrix $U_t \in \mathbb{R}^{n_\mu\times k_t}$, was zu

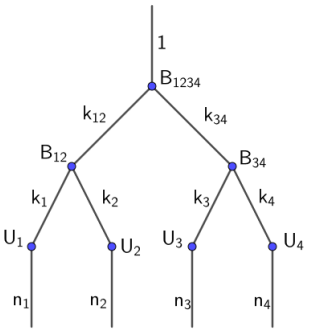
$$\text{Speicher}((U_t)_{t\in\mathcal{L}(T_I)}) \leqslant dnk \qquad n = \max_{\mu\in\{1,...,d\}}(n_\mu),\ k = \max_{t\in\mathcal{I}(T_I)} \tag{2.8}$$

führt. Aus (2.7) und (2.8) folgt nun (2.6). □

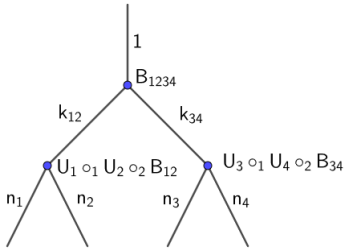
Die Abschätzung in (2.6) lässt einen linearen Zusammenhang zwischen Speicherbedarf eines hierarchischen Tuckertensors und der Anzahl an Dimensionen d vermuten. Dieser wünschenswerte Fall tritt allerdings nur unter der Voraussetzung ein, dass der maximale hierarchische Rang k bei anwachsendem d tatsächlich konstant bleibt. Abbildung 2.17 zeigt diagrammatisch, wie aus dem hierarchischen Tuckerformat eines Tensors, der volle Tensor wiederhergestellt werden kann.

2.2.3 Kürzungsoperationen

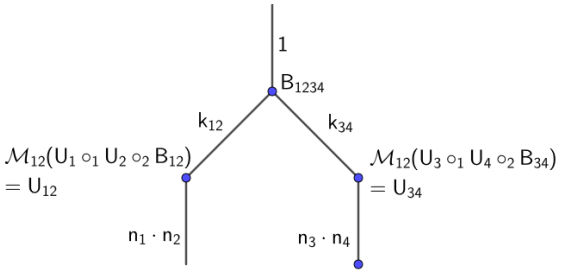
Dieser Abschnitt widmet sich den beiden Fragen, wie einerseits eine Repräsentation eines vollen Tensors im hierarchischen Tuckerformat berechnet werden kann und wie andererseits ein bereits bestehender hierarchischer Tuckertensor durch einen hierarchischen Tuckertensor mit kleinerem hierarchischen Rang approximiert werden kann.



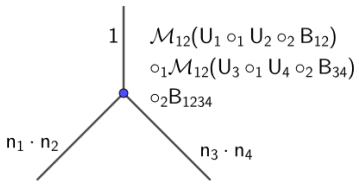
(a) Ausgangspunkt: X im hierarchischen Tuckerformat



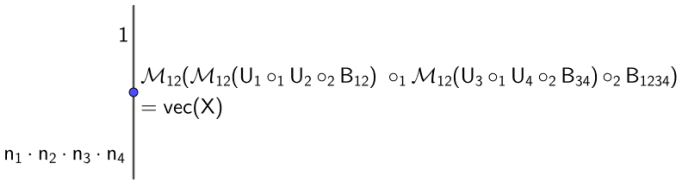
(b) Zustand nach den Kontraktionen auf Level 1



(c) Zustand nach den Matrizierungen auf Level 1



(d) Zustand nach den Kontraktionen auf Level 0



(e) Endpunkt: Zustand nach der Matrizierung auf Level 0

Abbildung 2.17: Schrittweise Darstellung der Wiederherstellung eines vollen Tensors X aus gegebenem hierarchischen Tuckerformat.

2.2.3.1 Kürzen voller Tensoren

In diesem Unterabschnitt erfolgt die Beschreibung des in [3] dargelegten Verfahrens, um aus einem vollen Tensor eine Repräsentation im hierarchischen Tuckerformat zu erhalten. Dieses basiert auf etappenweisen Projektionen auf die Unterräume der Spaltenräume der Matrizierungen des vollen Tensors.

Definition 2.40 (Orthogonale Projektion)
Sei $I = \{1, ..., n_1\} \times ... \times \{1, ..., n_d\}$ eine Indexmenge, $X \in \mathbb{R}^I$ ein reeller Tensor, $(k_t)_{t \in T_I}$ eine Rangverteilung auf T_I, $t \in T_I$ ein Moduscluster und $n_t = \prod_{\mu \in t} n_\mu$. Für eine Matrix

$U_t \in \mathbb{R}^{n_t \times k_t}$ mit orthonormalen Spalten und

$$\text{image}(U_t) \subseteq \text{image}(\mathcal{M}_t(X))$$

ist die orthogonale Projektion $\pi_t : \mathbb{R}^I \to \mathbb{R}^I$ in matrizierter Form durch

$$\mathcal{M}_t(\pi_t X) := U_t U_t^T \mathcal{M}_t(X), \quad \pi_\varnothing A := A, \quad \pi_{\{1,\dots,d\}} A := A$$

definiert.

Wendet man für jeden Moduscluster eine entsprechende orthogonale Projektion auf den vollen Tensor an, erhält man eine zugehörige Repräsentation im hierarchischen Tuckerformat mit einem hierarchischen Rang, der durch die orthogonalen Projektionen bestimmt ist. Wichtig dabei ist, dass die orthogonalen Projektionen nicht zwingend kommutieren: Die Level des Dimensionsbaums (siehe Def. 2.32) müssen mit der Wurzel beginnend von oben nach unten durchlaufen werden.

Definition 2.41 (Hierarchische Tuckerkürzung)
Seien $X \in \mathbb{R}^I$ ein reeller Tensor, T_I der zugehörige kanonische Dimensionsbaum mit Tiefe p und $(k_t)_{t \in T_I}$ eine Rangverteilung auf T_I. Dann ist die *hierarchische Tuckerkürzung* $X_\mathcal{H} \in \mathcal{H}\text{-Tucker}\left((k_t)_{t \in T_I}\right)$ als folgendes Produkt

$$X_\mathcal{H} := \prod_{t \in T_I^p} \pi_t \dots \prod_{t \in T_I^1} \pi_t X \tag{2.9}$$

gegeben. Grundlage der orthogonalen Projektionen π_t bilden hierbei die Matrizen U_t, deren Spalten den k_t dominanten linken Singulärvektoren von $\mathcal{M}_t(X)$ entsprechen.

Die Güte einer hierarchischen Tuckerkürzung kann hinsichtlich des Abschneidefehlers mit folgendem Theorem, das dafür eine obere Fehlerschranke angibt, bestimmt werden.

Theorem 2.42 (Hierarchischer Kürzungsfehler)
Seien $X \in \mathbb{R}^I$ ein reeller Tensor, T_I der kanonische Dimensionsbaum zur Indexmenge I und $(k_t)_{t \in T_I}$ eine Rangverteilung. Des Weiteren sei $X_\mathcal{H} \in \mathcal{H}\text{-Tucker}((k_t)_{t \in T_I})$ und $\sigma_{t,i}$ notiere den i-ten größten Singulärwert der Matrizierung $\mathcal{M}_t(X)$. Dann gilt die Abschätzung

$$||X - X_\mathcal{H}|| \leqslant \sqrt{\sum_{t \in T_I'} \sum_{i > k_t} \sigma_{t,i}^2} \leqslant \sqrt{2d-3}\, ||X - X^{\text{best}}||, \tag{2.10}$$

wobei X^{best} die beste Approximation von X in $\mathcal{H}\text{-Tucker}((k_t)_{t \in T_I})$ bezeichnet und T_I' als $T_I' := T_I \backslash \{t_{\text{Wurzel}}, t_{\text{Kind}}\}$ mit $t_{\text{Kind}} \in S(t_{\text{Wurzel}})$ gewählt ist.

Beweis. Das Theorem entspricht Theorem 17 und Remark 18 in [3]. □

Obiges Theorem kann umgekehrt auch dazu genutzt werden, um basierend auf (2.10) jenen hierarchischen Rang zu ermitteln, der notwendig ist, um eine gewünschte Fehlertoleranz einzuhalten.

Anmerkung 2.43
Ein Einhalten der absoluten Fehlerschranke ϵ_{abs} gemäß

$$||X - X_{\mathcal{H}}|| \leqslant \epsilon_{\text{abs}}$$

kann durch die Wahl eines hierarchischen Ranges $(k_t)_{t \in T_I}$ mit

$$\forall t \in T_I \backslash t_{\text{Wurzel}} : \sum_{i > k_t} \sigma_{t,i} \leqslant \frac{\epsilon_{\text{abs}}}{\sqrt{2d-3}}$$

erreicht werden.
Eine relative Fehlerschranke ϵ_{rel} entsprechend

$$||X - X_{\mathcal{H}}|| \leqslant \epsilon_{\text{rel}} ||X||$$

kann hingegen durch die Wahl eines hierarchischen Ranges $(k_t)_{t \in T_I}$ mit

$$\forall t \in T_I \backslash t_{\text{Wurzel}} : \sum_{i > k_t} \sigma_{t,i} \leqslant \frac{\epsilon_{\text{rel}} ||X||}{\sqrt{2d-3}}$$

eingehalten werden.

Unter Zuhilfenahme des Kronecker-Produkts präsentiert das folgende Beispiel die hierarchische Tuckerkürzung aus einem anderen Blickwinkel. Aufschlussreich ist hierbei ein Vergleich mit Beispiel 2.37 und Gleichung (2.2).

Beispiel 2.44
Sei X ein Tensor vierter Ordnung und $X_{\mathcal{H}}$ seine hierarchische Tuckerkürzung basierend auf den orthogonalen Projektionen $\pi_t = U_t U_t^T$. Dann kann die Vektorisierung von $X_{\mathcal{H}}$ auch wie folgt ausgedruckt werden:

$$\begin{aligned}
\text{vec}(X_{\mathcal{H}}) &= (U_4 U_4^T \otimes_{\mathcal{K}} U_3 U_3^T \otimes_{\mathcal{K}} U_2 U_2^T \otimes_{\mathcal{K}} U_1 U_1^T)(U_{34} U_{34}^T \otimes_{\mathcal{K}} U_{12} U_{12}^T)\text{vec}(X) \\
&= (U_4 \otimes_{\mathcal{K}} U_3 \otimes_{\mathcal{K}} U_2 \otimes_{\mathcal{K}} U_1)(U_4^T \otimes_{\mathcal{K}} U_3^T \otimes_{\mathcal{K}} U_2^T \otimes_{\mathcal{K}} U_1^T)(U_{34} \otimes_{\mathcal{K}} U_{12})(U_{34}^T \otimes_{\mathcal{K}} U_{12}^T)\text{vec}(X) \\
&= (U_4 \otimes_{\mathcal{K}} U_3 \otimes_{\mathcal{K}} U_2 \otimes_{\mathcal{K}} U_1)(\underbrace{(U_4^T \otimes_{\mathcal{K}} U_3^T)U_{34}}_{=B_{34}} \otimes_{\mathcal{K}} \underbrace{(U_2^T \otimes_{\mathcal{K}} U_1^T)U_{12}}_{=B_{12}}) \underbrace{(U_{34}^T \otimes_{\mathcal{K}} U_{12}^T)\text{vec}(X)}_{=B_{1234}}
\end{aligned}$$

Algorithmus 1 fasst die Berechnung der hierarchischen Tuckerkürzung zusammen. Hervorzuheben ist hierbei, dass der resultierende hierarchische Tuckertensor kein orthogonaler hierarchischer Tuckertensor ist.

Algorithm 1: Hierarchische Tuckerkürzung

Input: Tensor $X \in \mathbb{R}^I$, Kanonischer Dimensionsbaum T_I (mit Tiefe p), hierarchischer Rang $(k_t)_{t \in T_I}$

Output: $X_{\mathcal{H}} \in \mathcal{H}\text{-Tucker}((k_t)_{t \in T_I})$ im hierarchischen Tuckerformat $\left((U_t)_{t \in \mathcal{L}(T_I)}, (B_t)_{t \in \mathcal{I}(T_I)}\right)$

```
/* Als erstes werden die Blattknoten durchschritten                    */
```
for *each* $t \in \mathcal{L}(T_I)$ **do**
  Berechne SVD $\hat{U}_t \hat{\Sigma}_t \hat{V}_t \leftarrow \mathcal{M}_t(X)$
  $U_t \leftarrow \hat{U}_t[:, : k_t]$ // Übernahme der ersten k_t Spalten

```
/* Nun werden von unten nach oben alle inneren Knoten außer der Wurzel durchlaufen
   */
```
for $l = p - 1, ..., 1$ **do**
  for *each* $t \in T_I^l$ **do**
    Berechne SVD $\hat{U}_t \hat{\Sigma}_t \hat{V}_t \leftarrow \mathcal{M}_t(X)$
    $U_t \leftarrow \hat{U}_t[:, : k_t]$ // Übernahme der ersten k_t Spalten
    Seien t_1 und t_2 die Kinder von t auf Level $l + 1$
    $B_t \leftarrow (U_{t_2} \otimes_{\mathcal{K}} U_{t_1}) U_t$

```
/* Zum Abschluss wird der Wurzelknoten behandelt                       */
```
Seien t_1 und t_2 die Kinder der Wurzel auf Level 1
$B_{t_{\text{Wurzel}}} \leftarrow (U_{t_2} \otimes_{\mathcal{K}} U_{t_1}) \text{vec}(X)$

Lemma 2.45 (Laufzeitkomplexität von Algorithmus 1)
Sei $X \in \mathbb{R}^I$ mit Indexmenge $I = \{1, ..., n_1\} \times ... \times \{1, ..., n_d\}$ und Dimensionsbaum T_I samt Tiefe $p > 0$. Dann liegt die Laufzeitkomplexität von Algorithmus 1 in $\mathcal{O}\left(\left(\prod_{\mu=1}^{d} n_\mu\right)^{\frac{3}{2}}\right)$.

Der Beweis zu Lemma 2.45 findet sich in [3] als Beweis des 24. Lemmas, soll hier aber trotzdem wiederholt werden.

Beweis. Während der hierarchischen Tuckerkürzung müssen Singulärwertzerlegungen zu den verschiedenen Matrizierungen $\mathcal{M}_t(X)$ von X berechnet werden. Diese Berechnungen haben jeweils eine Laufzeitkomplexität in $\mathcal{O}\left(\min(\prod_{\mu \in t} n_\mu, \prod_{\mu \in [t]} n_\mu)^2 \max(\prod_{\mu \in t} n_\mu, \prod_{\mu \in [t]} n_\mu)\right)$, wobei $t \in T_I$ und $[t] := \{1, ..., d\} \backslash t$. Damit hat die Wurzel eine Laufzeitkomplexität von null und unter der nicht die Allgemeinheit einschränkenden Annahme, dass für alle Modi $n_\mu \geqslant 2$ gilt, folgt für die Kinder t und $[t]$ der Wurzel eine Laufzeitkomplexität gemäß

$$\mathcal{C}_{\text{SVD}} \left(\min(\prod_{\mu \in t} n_\mu, \prod_{\mu \in [t]} n_\mu)^2 \max(\prod_{\mu \in t} n_\mu, \prod_{\mu \in [t]} n_\mu) \right) \leqslant \mathcal{C}_{\text{SVD}} \left(\prod_{\mu=1}^{d} n_\mu \right)^{\frac{3}{2}},$$

wobei $\mathcal{C}_{\text{SVD}}$ eine allgemeine Konstante der Singulärwertzerlegung darstellt. Für jedes weitere

Level werden $\prod_{\mu \in t} n_\mu$ und $\prod_{\mu \in [t]} n_\mu$ (wegen $\forall \mu \in \{1, ..., d\} : n_\mu \geqslant 2$) um mindestens Faktor zwei reduziert, sodass sich die Laufzeitkomplexität der Singulärwertzerlegungen insgesamt viertelt, während sich die Anzahl der Modi höchstens verdoppelt. Damit ist die gesamte Laufzeitkomplexität beschränkt durch

$$\sum_{l=0}^{p} 2^{-l} \mathcal{C}_{\text{SVD}} \left(\prod_{\mu=1}^{d} n_\mu \right)^{\frac{3}{2}} \leqslant 2\, \mathcal{C}_{\text{SVD}} \left(\prod_{\mu=1}^{d} n_\mu \right)^{\frac{3}{2}}.$$

□

Der Beweis zeigt, dass die Laufzeitkomplexität durch die kostspieligen Berechnungen der Singulärwertzerlegungen der verschiedenen Matrizierungen des vollständigen Tensors dominiert wird. Große Matrizierungen können damit schwierig handhabbar werden. Ein Algorithmus, der dieses Problem ein wenig entschärft, berechnet nicht die Singulärwertzerlegungen des vollständigen Tensors, sondern die eines Kerntensors, der jedes Level kleiner wird (siehe Algorithmus 2 in [3] oder Algorithmus 5 in [11]).

2.2.3.2 Kürzung von Tensoren im hierarchischen Tuckerformat

Einige Rechenoperationen im hierarchischen Tuckerformat führen zu einem Anwachsen des hierarchischen Ranges. Da sowohl die Laufzeitkomplexität der Rechenoperationen als auch der Speicherplatz vom hierarchischen Rang abhängen, kann es notwendig werden, einen hierarchischen Tuckertensor durch einen hierarchischen Tuckertensor mit niedrigerem hierarchischen Rang zu approximieren. Das in [11] präsentierte Vorgehen dazu ist im Wesentlichen eine Adaption von Algorithmus 1, basierend auf reduzierten Gram'schen Matrizen und dem orthogonalen hierarchischen Tuckerformat (siehe Definition 2.38).

Definition 2.46 (Reduzierte Gram'sche Matrix)
Sei T_I ein kanonischer Dimensionsbaum, $(k_t)_{t \in T_I}$ eine Rangverteilung und ferner $X \in \mathcal{H}\text{-Tucker}((k_t)_{t \in T_I})$ ein reeller Tensor mit hierarchischem Tuckerformat $((U_t)_{t \in \mathcal{L}(T_I)}, (B_t)_{t \in \mathcal{I}(T_I)})$. Dann enthält U_t eine Basis für die Matrizierung $\mathcal{M}_t(X)$ und entsprechend existiert eine Matrix V_t mit $\mathcal{M}_t(X) = U_t V_t^T$. Die *reduzierte Gram'sche Matrix* in t ist dann zu definieren als die symmetrische positiv semidefinite Matrix

$$G_t := V_t^T V_t \in \mathbb{R}^{k_t \times k_t}.$$

Das Vorgehen zum Kürzen eines hierarchischen Tuckertensors beginnt im ersten Schritt mit dessen Orthogonalisieren (siehe dazu Algorithmus 1 in [11]), sofern er nicht bereits in orthogonalisierter Form vorliegt. Anschließend werden die reduzierten Gram'schen Matrizen G_t für jeden Moduscluster bestimmt. Ein entsprechendes Verfahren dazu findet sich unter

Algorithmus 3 in [11]. Sei also

$$G_t = Q_t \Lambda Q_t^T$$

eine Spektralzerlegung der reduzierten Gram'schen Matrix G_t. Dann stimmen einerseits die Eigenvektoren von G_t, gegeben als Spalten der Matrix Q_t, mit den linken Singulärvektoren von V_t^T überein und andererseits folgt aus

$$\mathcal{M}_t(X)\mathcal{M}_t(X)^T = U_t V_t^T V_t U_t^T = U_t G_t U_t^T,$$

dass die Singulärwerte von $\mathcal{M}_t(X)$ den Quadratwurzeln der Eigenwerte von G_t entsprechen, sofern $U_t^T U_t = I_{k_t}$ gilt. Wegen des vorherigen Orthogonalisierens ist diese Voraussetzung jedoch erfüllt. Die Spalten des Produkts $U_t Q_t$ sind damit linke Singulärvektoren von $\mathcal{M}_t(X)$ deren zugehörige Singulärwerte mit den Quadratwurzeln der Eigenwerte von G_t übereinstimmen. Für einen neuen angestrebten Rang von r_t im Moduscluster t werden nur die dominanten r_t linken Singulärvektoren beibehalten. Aufbauend darauf werden orthogonale Projektionen π_t auf die zugehörigen Unterräume durch $\mathcal{M}_t(\pi_t X) := U_t Q_t (U_t Q_t)^T \mathcal{M}_t(X)$ definiert. Das weitere Vorgehen verläuft ab diesem Punkt analog zur hierarchischen Tuckerkürzung.

Algorithm 2: Kürzung von Tensoren im hierarchischen Tuckerformat

Input: $X \in \mathcal{H}\text{-Tucker}((k_t)_{t\in T_I})$ im hierarchischen Tuckerformat $\left((U_t)_{t\in\mathcal{L}(T_I)}, (B_t)_{t\in\mathcal{I}(T_I)}\right)$, neuer hierarchischer Rang $(r_t)_{t\in T_I}$

Output: $\tilde{X} \in \mathcal{H}\text{-Tucker}((r_t)_{t\in T_I})$ im hierarchischen Tuckerformat $\left((\tilde{U}_t)_{t\in\mathcal{L}(T_I)}, (\tilde{B}_t)_{t\in\mathcal{I}(T_I)}\right)$

Orthogonalisiere X
Berechne die reduzierten Gram'schen Matrizen G_t für alle $t \in T_I$
for *each* $t \in T_I \backslash \{t_{Wurzel}\}$ **do**
    Berechne Spektralzerlegung $Q\Lambda Q^T \leftarrow G_t$
    $Q_t \leftarrow Q[:, : r_t]$ `// Übernahme der ersten` r_t `Spalten`
$Q_{t_{\text{Wurzel}}} \leftarrow 1$
`/* Updaten der Blattknoten */`
for *each* $t \in \mathcal{L}(T_I)$ **do**
    $\tilde{U}_t \leftarrow U_t Q_t$
`/* Updaten der inneren Knoten */`
for *each* $t \in \mathcal{I}(T_I)$ **do**
    Seien t_1 und t_2 die Kinder von t
    $\tilde{B}_t \leftarrow (Q_{t_2}^T \otimes_{\mathcal{K}} Q_{t_1}^T) B_t Q_t$

Durch die mathematische Übereinstimmung mit Algorithmus 1 kann entsprechend Anmerkung 2.43 der hierarchische Rang so bestimmt werden, dass bei der Kürzung eine vor-

gegebene Fehlertoleranz eingehalten wird.

Lemma 2.47 (Laufzeitkomplexität von Algorithmus 2)
Sei X der zu kürzende Tensor im hierarchischen Tuckerformat mit $X \in \mathcal{H}\text{-Tucker}((k_t)_{T_I})$ und Indexmenge $I = \{1, ..., n_1\} \times ... \times \{1, ..., n_d\}$. Dann hat Algorithmus 2 eine Laufzeitkomplexität von $\mathcal{O}\left(dnk^2 + dk^4\right)$ mit $n = \max\limits_{\mu \in \{1,...,d\}}(n_\mu)$ und $k = \max\limits_{t \in T_I}(k_t)$.

Beweis. Die Laufzeitkomplexität findet sich unter [11] im Abschnitt zu Algorithmus 6. □

2.2.4 Rechenoperationen im hierarchischen Tuckerformat

Im vorherigen Abschnitt wurde dargelegt, wie ein voller Tensor in das hierarchische Tuckerformat überführt und approximiert werden kann. In diesem Abschnitt soll wiederum erklärt werden, wie grundlegende Tensoroperationen (vgl. Abschnitt 2.1.3) direkt im hierarchischen Tuckerformat durchgeführt werden können. Die Ausführungen zur Skalarmultiplikation, Addition und elementweisen Multiplikation basieren dabei auf [11], wobei die Bereitstellung von Pseudocode für die Addition und elementweise Multiplikation über diese Grundlage hinausgeht. Wenngleich die Indexkontraktion hierarchischer Tuckertensoren in diesem Abschnitt nicht erläutert wird, soll dennoch hervorgehoben werden, dass diese prinzipiell möglich ist. Wie sie implementiert werden kann und mit welchen Einschränkungen sie einhergeht, findet sich ebenfalls in [11].

2.2.4.1 Skalarmultiplikation

Die Multiplikation eines Tensors $X \in \mathcal{H}\text{-Tucker}((k_t)_{t \in T_I})$ im hierarchischen Tuckerformat $\left((U_t)_{t \in \mathcal{L}(T_I)}, (B_t)_{t \in \mathcal{I}(T_I)}\right)$ mit einem Skalar $\lambda \in \mathbb{R}$ erfolgt durch Multiplikation des Transfertensors der Wurzel $B_{t_{\text{Wurzel}}}$ mit λ.

Beispiel 2.48
Sei X ein Tensor vierter Ordnung mit hierarchischem Tuckerformat $\left((U_t)_{t \in \mathcal{L}(T_I)}, (B_t)_{t \in \mathcal{I}(T_I)}\right)$ und $\lambda \in \mathbb{R}$. Dann gilt

$$\text{vec}(X) = (U_4 \otimes_{\mathcal{K}} U_3 \otimes_{\mathcal{K}} U_2 \otimes_{\mathcal{K}} U_1)(B_{34} \otimes_{\mathcal{K}} B_{12})B_{1234},$$

wodurch die Skalarmultiplikation mit λ durch

$$\begin{aligned}\lambda\text{vec}(X) &= \lambda(U_4 \otimes_{\mathcal{K}} U_3 \otimes_{\mathcal{K}} U_2 \otimes_{\mathcal{K}} U_1)(B_{34} \otimes_{\mathcal{K}} B_{12})B_{1234} \\ &= (U_4 \otimes_{\mathcal{K}} U_3 \otimes_{\mathcal{K}} U_2 \otimes_{\mathcal{K}} U_1)(B_{34} \otimes_{\mathcal{K}} B_{12}) \underbrace{\lambda B_{1234}}_{=:B^{\text{neu}}_{1234}}\end{aligned}$$

gegeben ist.

2.2.4.2 Addition

Die Addition zweier Tensoren im hierarchischen Tuckerformat ist eine grundlegende Operation, die keinerlei Rechenaufwand benötigt, jedoch voraussetzt, dass die Dimensionsbäume beider Summanden übereinstimmen.
Seien X und Y zwei Matrizen mit Singulärwertzerlegungen $X = U^x\Sigma^x(V^x)^T$ beziehungsweise $Y = U^y\Sigma^y(V^y)^T$. Dann ist ihre Summe darstellbar als

$$\begin{aligned} X + Y &= U^x\Sigma^x(V^x)^T + U^y\Sigma^y(V^y)^T \\ &= \begin{bmatrix} U^x & U^y \end{bmatrix} \begin{bmatrix} \Sigma^x & 0 \\ 0 & \Sigma^y \end{bmatrix} \begin{bmatrix} V^x & V^y \end{bmatrix}^T. \end{aligned}$$

Verallgemeinert auf das hierarchische Tuckerformat, folgt, dass die Blattmatrizen der Summanden pro Blattknoten konkateniert werden, während die Transfertensoren pro innerem Knoten in einen umspannenden blockdiagonalen Transfertensor eingefasst werden (siehe Abbildung 2.18 und Algorithmus 3). Eine Ausnahme bildet hierbei der Wurzelknoten, da die Transfertensoren dort zu einer blockdiagonalen Matrix geformt werden.

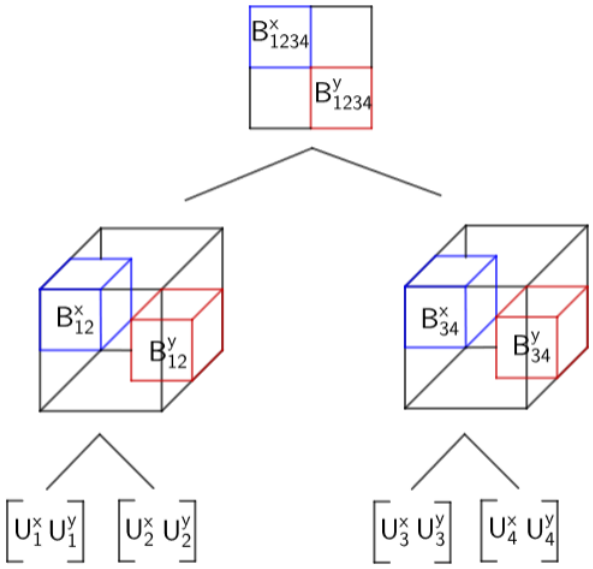


Abbildung 2.18: Addition zweier Tensoren der Ordnung vier im hierarchischen Tuckerformat.

Aus diesem Vorgehen folgt, dass der resultierende hierarchische Rang $(k_t)_{t\in T_I}$ durch

$$\forall t \in T_I : k_t = k_t^x + k_t^y$$

gegeben ist, was bei übereinstimmenden hierarchischen Rängen der Summanden einer Ver-

dopplung gleichkommt. Werden mehrere Additionen ausgeführt, ist es damit nahezu unvermeidbar, eine anschließende Rangkürzung vorzunehmen. Da deren Laufzeit jedoch vom maximalen hierarchischen Rang k gemäß $\mathcal{O}\left(dnk^2 + dk^4\right)$ abhängt, kann dies sehr kostspielig werden. Abhilfe verspricht ein kombiniertes Verfahren aus Rangkürzung und Addition (siehe Abschnitt 6.3 in [11]).

Algorithm 3: Addition von Tensoren im hierarchischen Tuckerformat

Input: $X \in \mathcal{H}\text{-Tucker}((k_t^x)_{t\in T_I})$ im hierarchischen Tuckerformat $\left((U_t^x)_{t\in\mathcal{L}(T_I)}, (B_t^x)_{t\in\mathcal{I}(T_I)}\right)$, $Y \in \mathcal{H}\text{-Tucker}((k_t^y)_{t\in T_I})$ im hierarchischen Tuckerformat $\left((U_t^y)_{t\in\mathcal{L}(T_I)}, (B_t^y)_{t\in\mathcal{I}(T_I)}\right)$

Output: $X + Y \in \mathcal{H}\text{-Tucker}((k_t^x + k_t^y)_{t\in T_I})$ im hierarchischen Tuckerformat $\left((U_t)_{t\in\mathcal{L}(T_I)}, (B_t)_{t\in\mathcal{I}(T_I)}\right)$

```
/* Konkatenieren der Blattmatrizen                                   */
```

for *each* $t \in \mathcal{L}(T_I)$ **do**

  $U_t \leftarrow \begin{bmatrix} U_t^x & U_t^y \end{bmatrix}$ // $U_t \in \mathbb{R}^{n_t \times (k_t^x + k_t^y)}$

```
/* Einbetten der Transfertensoren                                    */
```

for *each* $t \in \mathcal{I}(T_I)\backslash\{t_{Wurzel}\}$ **do**

  Seien t_1 und t_2 die Kinder von t

  Sei B_t ein $(k_{t_1}^x + k_{t_1}^y) \times (k_{t_2}^x + k_{t_2}^y) \times (k_t^x + k_t^y)$ Tensor bestehend nur aus Nullen

  $B_t[: k_{t_1}^x, : k_{t_2}^x, : k_t^x] \leftarrow B_t^x$

  $B_t[k_{t_1}^x :, k_{t_2}^x :, k_t^x :] \leftarrow B_t^y$

```
/* Einbetten der Wurzel-Transfertensoren                             */
```

Seien t_1 und t_2 die Kinder von t_{Wurzel}

Sei $B_{t_{\text{Wurzel}}}$ eine $(k_{t_1}^x + k_{t_1}^y) \times (k_{t_2}^x + k_{t_2}^y)$ Matrix bestehend nur aus Nullen

$B_{t_{\text{Wurzel}}}[: k_{t_1}^x, : k_{t_2}^x] \leftarrow B_{t_{\text{Wurzel}}}^x$

$B_{t_{\text{Wurzel}}}[k_{t_1}^x :, k_{t_1}^x :] \leftarrow B_{t_{\text{Wurzel}}}^y$

2.2.4.3 Modusmultiplikation

Im hierarchischen Tuckerformat ist die Modusmultiplikation mit geringen Laufzeitkosten verbunden und führt anders als die Addition nicht zu einem Anwachsen des hierarchischen Ranges. Für einen Tensor $X \in \mathbb{R}^{n_1\times\ldots\times n_d}$ lässt sich die Modusmultiplikation mit einer Matrix $A \in \mathbb{R}^{m\times n_\mu}$ via Modus μ durch folgende Vektorisierung ausdrücken:

$$\operatorname{vec}(A \circ_\mu X) = \left(I_{n_d} \otimes_{\mathcal{K}} \ldots \otimes_{\mathcal{K}} I_{n_{\mu+1}} \otimes_{\mathcal{K}} A \otimes_{\mathcal{K}} I_{n_{\mu-1}} \otimes_{\mathcal{K}} \ldots \otimes_{\mathcal{K}} I_{n_1}\right) \operatorname{vec}(X)$$

Zusammen mit der Repräsentation von X im hierarchischen Tuckerformat gegeben als

$$\operatorname{vec}(X) = (U_d \otimes_{\mathcal{K}} \ldots \otimes_{\mathcal{K}} U_1) \ldots \left(B_{\frac{d}{2}+1\ldots d} \otimes_{\mathcal{K}} B_{1\ldots\frac{d}{2}}\right) B_{1\ldots d}$$

ergibt sich schließlich

$$\begin{aligned}\operatorname{vec}(A \circ_\mu X) &= \left(I_{n_d} \otimes_{\mathcal{K}} \ldots \otimes_{\mathcal{K}} I_{n_{\mu+1}} \otimes_{\mathcal{K}} A \otimes_{\mathcal{K}} I_{n_{\mu-1}} \otimes_{\mathcal{K}} \ldots \otimes_{\mathcal{K}} I_{n_1}\right)(U_d \otimes_{\mathcal{K}} \ldots \otimes_{\mathcal{K}} U_1) \\ &\quad \ldots \left(B_{d/2+1\ldots d} \otimes_{\mathcal{K}} B_{1\ldots d/2}\right) B_{1\ldots d} \\ &= (U_d \otimes_{\mathcal{K}} \ldots \otimes_{\mathcal{K}} AU_\mu \otimes_{\mathcal{K}} \ldots \otimes_{\mathcal{K}} U_1) \ldots \left(B_{d/2+1\ldots d} \otimes_{\mathcal{K}} B_{1\ldots d/2}\right) B_{1\ldots d}.\end{aligned}$$

Dementsprechend wird die Modusmultiplikation im hierarchischen Tuckerformat durch Anpassen der Blattmatrix U_μ gemäß

$$U_\mu^{\text{neu}} := AU_\mu \in \mathbb{R}^{m \times k_\mu}$$

erreicht.

2.2.4.4 Elementweise Multiplikation

Die Berechnung des elementweisen Produkts zweier Tensoren im hierarchischen Tuckerformat kann entweder exakt oder im Verbund mit einer Approximation ausgeführt werden. Einleitend soll beispielhaft gezeigt werden, wie sich die elementweise Multiplikation zweier Matrizen durch deren Singulärwertzerlegungen ausdrücken lässt.
Seien also zwei Matrizen X und Y mit gleich vielen Zeilen und Spalten inklusive Singulärwertzerlegungen

$$\begin{aligned}X &= U^x \Sigma^x (V^x)^T = \sum_i \sigma_i^x u_i^x (v_i^x)^T, \\ Y &= U^y \Sigma^y (V^y)^T = \sum_i \sigma_i^y u_i^y (v_i^y)^T\end{aligned}$$

gegeben. Dann lässt sich das elementweise Produkt $(X \star Y)$ wie folgt ausdrücken:

$$\begin{aligned}(X \star Y) &= \sum_i \sigma_i^x u_i^x (v_i^x)^T \star \sum_j \sigma_j^y u_j^y (v_i^y)^T \\ &= \sum_{i,j} \sigma_i^x \sigma_j^y (u_i^x (v_i^x)^T) \star (u_j^y (v_j^y)^T) \\ &= \sum_{i,j} (u_i^x \star u_j^y)(\sigma_i^x \sigma_j^y)((v_i^x)^T \star (v_j^y)^T) \\ &= (U^x \odot^T U^y)(\Sigma^x \otimes_{\mathcal{K}} \Sigma^y)(V^x \odot^T V^x)^T\end{aligned}$$

Wobei mit $\odot^T$ das transponierte Khatri-Rao Produkt anzeigt wird (siehe Anhang A.3). Erweitert man die Definition des Kronecker-Produkts auf Tensoren (siehe Anhang A.4), können die Blattmatrizen und Transfertensoren des elementweisen Produkts zweier Tenso-

ren X und Y im hierarchischen Tuckerformat analog dazu wie folgt berechnet werden:

$$\forall t \in \mathcal{L}(T_I) : U_t := U_t^x \odot^T U_t^y$$
$$\forall t \in \mathcal{I}(T_I) : B_t := B_t^x \otimes_{\mathcal{K}} B_t^y$$

Dieses exakte Vorgehen führt zu einem hierarchischen Rang $(k_t)_{t \in T_I}$ des resultierenden hierarchischen Tuckertensors mit

$$\forall t \in T_I : k_t = k_t^x k_t^y.$$

Es wäre hilfreich, dieses quadratische Anwachsen des Ranges bereits bei der Berechnung der neuen Blattmatrizen und Transfertensoren durch eine Rangkürzung einzuschränken. Anders als bei der Addition ist allerdings nicht bekannt, wie die beiden Vorgänge direkt miteinander verknüpft werden können. Die beiden folgenden Beobachtungen eröffnen über einen Umweg die Möglichkeit, das Rangwachstum dennoch etwas einzuhegen.
Die erste Beobachtung besteht darin, dass das elementweise Produkt zweier Tensoren in deren Kronecker-Produkt enthalten ist. Konkret existieren Matrizen $J_{n_\mu} \in \mathbb{R}^{n_\mu \times n_\mu^2}$ mit

$$J_{n_\mu}[i,j] := \begin{cases} 1 & j = (i-1)n_\mu + i \\ 0 & \text{sonst} \end{cases},$$

sodass für $X, Y \in \mathbb{R}^{n_1 \times \ldots \times n_d}$ gilt:

$$\operatorname{vec}(X \star Y) = (J_{n_d} \otimes_{\mathcal{K}} \ldots \otimes_{\mathcal{K}} J_{n_1}) \operatorname{vec}(X \otimes_{\mathcal{K}} Y)$$

Die zweite Beobachtung bezieht sich wiederum auf die Möglichkeit, das Kronecker-Produkt zweier Tensoren im hierarchischen Tuckerformat zu berechnen und gleichzeitig den hierarchischen Rang zu kürzen. Da die Blattmatrizen und Transfertensoren des Kronecker-Produkts zweier Tensoren X und Y im hierarchischen Tuckerformat durch

$$\forall t \in \mathcal{L}(T_I) : U_t := U_t^x \otimes_{\mathcal{K}} U_t^y$$
$$\forall t \in \mathcal{I}(T_I) : B_t := B_t^x \otimes_{\mathcal{K}} B_t^y$$

gegeben sind, haben die reduzierten Gram'schen Matrizen ebenfalls Kronecker-Produkt-Struktur:

$$\forall t \in T_I : G_t = G_t^x \otimes_{\mathcal{K}} G_t^y$$

Daraus folgt wiederum, dass auch die Singulärwertzerlegungen der reduzierten Gram'schen Matrizen Kronecker-Produkt-Struktur aufweisen, sodass die linken Singulärvektoren samt

zugehöriger Singulärwerte effizient berechnet werden können. Demzufolge kann sogar das übliche Kürzungsverfahren für Tensoren im hierarchischen Tuckerformat (siehe Algorithmus 2) inklusive vorgegebener Fehlertoleranz verwendet werden. Wenngleich der hierarchische Rang des resultierenden hierarchischen Tuckertensor mit diesem Vorgehen in der Regel merkbar kleiner ausfällt, ist es trotzdem ratsam, im Anschluss eine weitere Kürzung gemäß Algorithmus 2 durchzuführen, da die erhaltene Fehlerschranke in der Regel eher großzügig ausfällt.

Algorithm 4: Elementweise Multiplikation von Tensoren im hierarchischen Tuckerformat

Input: $X \in \mathcal{H}\text{-Tucker}((k_t^x)_{t\in T_I})$ im hierarchischen Tuckerformat $\left((U_t^x)_{t\in\mathcal{L}(T_I)}, (B_t^x)_{t\in\mathcal{I}(T_I)}\right)$, $Y \in \mathcal{H}\text{-Tucker}((k_t^y)_{t\in T_I})$ im hierarchischen Tuckerformat $\left((U_t^y)_{t\in\mathcal{L}(T_I)}, (B_t^y)_{t\in\mathcal{I}(T_I)}\right)$, neuer hierarchischer Rang $(r_t)_{t\in T_I}$

Output: $X \star Y \in \mathcal{H}\text{-Tucker}((r_t)_{t\in T_I})$ im hierarchischen Tuckerformat $\left((U_t)_{t\in\mathcal{L}(T_I)}, (B_t)_{t\in\mathcal{I}(T_I)}\right)$

Orthogonalisiere X und Y
Berechne die reduzierten Gram'schen Matrizen G_t^x und G_t^y für alle $t \in T_I$
for *each* $t \in T_I \backslash \{t_{Wurzel}\}$ **do**
  Berechne Spektralzerlegung $Q^x \Lambda^x (Q^x)T \leftarrow G_t^x$
  Berechne Spektralzerlegung $Q^y \Lambda^y (Q^y)^T \leftarrow G_t^y$
  Seien $\sigma_{i_1}^x \sigma_{j_1}^y, \ldots, \sigma_{i_{r_t}}^x \sigma_{j_{r_t}}^y$ die r_t dominanten Singulärwertprodukte aus $\Lambda^x \otimes_{\mathcal{K}} \Lambda^y$
  $Q_t^x \leftarrow Q^x[:, (i_1, \ldots, i_{r_t})]$ // Übernahme der Spalten $i_1, \ldots, i_{r_t}$
  $Q_t^y \leftarrow Q^y[:, (j_1, \ldots, j_{r_t})]$ // Übernahme der Spalten $j_1, \ldots, j_{r_t}$
$Q_{t_{\text{Wurzel}}}^x \leftarrow 1$
$Q_{t_{\text{Wurzel}}}^y \leftarrow 1$
/* Berechnung der Blattmatrizen */
for *each* $t \in \mathcal{L}(T_I)$ **do**
  $\tilde{U}_t^x \leftarrow U_t^x Q_t^x$
  $\tilde{U}_t^y \leftarrow U_t^y Q_t^y$
  $U_t \leftarrow \tilde{U}_t^x \star \tilde{U}_t^y$
/* Berechnung der Transfertensoren */
for *each* $t \in \mathcal{I}(T_I)$ **do**
  Seien t_1 und t_2 die Kinder von t
  $\tilde{B}_t^x \leftarrow \left((Q_{t_2}^x)^T \otimes_{\mathcal{K}} (Q_{t_1}^x)^T\right) B_t^x Q_t^x$
  $\tilde{B}_t^y \leftarrow \left((Q_{t_2}^y)^T \otimes_{\mathcal{K}} (Q_{t_1}^y)^T\right) B_t^y Q_t^y$
  $B_t \leftarrow \tilde{B}_t^x \star \tilde{B}_t^y$

2.2.4.5 Elementweise Modusmultiplikation

Obwohl die elementweise Modusmultiplikation gewissermaßen einen Spezialfall der regulären elementweisen Multiplikation zweier Tensoren darstellt, lässt sie sich im hierarchischen

Tuckerformat berechnen, ohne den hierarchischen Rang zu erhöhen. Damit sollte sie im Kontext des hierarchischen Tuckerformats wann immer möglich der regulären elementweisen Multiplikation vorgezogen werden.
Sind ein beliebiger Tensor $X \in \mathbb{R}^{n_1 \times \ldots \times n_d}$ und ein Vektor $v \in \mathbb{R}^{n_\mu}$ mit $\mu \in \{1, \ldots, d\}$ gegeben, so lässt sich die Vektorisierung der elementweisen Modusmultiplikation $v \star_\mu X$ wie folgt darstellen:

$$\begin{aligned}\operatorname{vec}(v \star_\mu X) &= \operatorname{vec}(\mathbf{1}_{n_1} \otimes \ldots \otimes \mathbf{1}_{n_{\mu-1}} \otimes v \otimes \mathbf{1}_{n_{\mu+1}} \otimes \ldots \otimes \mathbf{1}_{n_d}) \star_1 \operatorname{vec}(X) \\ &= (\mathbf{1}_{n_d} \otimes_{\mathcal{K}} \ldots \otimes_{\mathcal{K}} \mathbf{1}_{n_{\mu+1}} \otimes_{\mathcal{K}} v \otimes \mathbf{1}_{n_{\mu-1}} \otimes_{\mathcal{K}} \ldots \otimes_{\mathcal{K}} \mathbf{1}_{n_1}) \star_1 \operatorname{vec}(X)\end{aligned}$$

Wobei mit $\mathbf{1}_{n_i}$ Vektoren bestehend aus n_i Einsen bezeichnet werden: $\mathbf{1}_{n_i} = (1, \ldots, 1)^T \in \mathbb{R}^{n_i}$. Zusätzlich wurde ausgenutzt, dass für Tensoren erster Ordnung die elementweise Modusmultiplikation entlang des ersten und einzigen Modus mit der regulären elementweisen Multiplikation übereinstimmt. In Kombination mit der Repräsentation von X im hierarchischen Tuckerformat

$$\operatorname{vec}(X) = (U_d \otimes_{\mathcal{K}} \ldots \otimes_{\mathcal{K}} U_1) \ldots \left(B_{\frac{d}{2}+1\ldots d} \otimes_{\mathcal{K}} B_{1\ldots\frac{d}{2}}\right) B_{1\ldots d}$$

folgt:

$$\begin{aligned}\operatorname{vec}(v \star_\mu X) &= (\mathbf{1}_{n_d} \otimes_{\mathcal{K}} \ldots \otimes_{\mathcal{K}} \mathbf{1}_{n_{\mu+1}} \otimes_{\mathcal{K}} v \otimes \mathbf{1}_{n_{\mu-1}} \otimes_{\mathcal{K}} \ldots \otimes_{\mathcal{K}} \mathbf{1}_{n_1}) \star_1 (U_d \otimes_{\mathcal{K}} \ldots \otimes_{\mathcal{K}} U_1) \\ &\quad \ldots \left(B_{\frac{d}{2}+1\ldots d} \otimes_{\mathcal{K}} B_{1\ldots\frac{d}{2}}\right) B_{1\ldots d} \\ &= (\mathbf{1}_{n_d} \star_1 U_d \otimes_{\mathcal{K}} \ldots \otimes_{\mathcal{K}} v \star_1 U_\mu \otimes_{\mathcal{K}} \ldots \otimes_{\mathcal{K}} \mathbf{1}_{n_1} \star_1 U_1) \ldots \left(B_{\frac{d}{2}+1\ldots d} \otimes_{\mathcal{K}} B_{1\ldots\frac{d}{2}}\right) B_{1\ldots d} \\ &= (U_d \otimes_{\mathcal{K}} \ldots \otimes_{\mathcal{K}} v \star_1 U_\mu \otimes_{\mathcal{K}} \ldots \otimes_{\mathcal{K}} U_1) \ldots \left(B_{\frac{d}{2}+1\ldots d} \otimes_{\mathcal{K}} B_{1\ldots\frac{d}{2}}\right) B_{1\ldots d}\end{aligned}$$

Demzufolge kann für einen Vektor v und einen Tensor X mit Repräsentation im hierarchischen Tuckerformat $\left((U_t)_{t\in\mathcal{L}(T_I)}, (B_t)_{t\in\mathcal{I}(T_I)}\right)$ die elementweise Modusmultiplikation $v \star_\mu X$ durch Anpassen der Blattmatrix U_μ gemäß

$$U_\mu^{neu} := v \star_1 U_\mu \in \mathbb{R}^{n_\mu \times k_\mu}$$

berechnet werden.

3 Epidemiologische Modelle

Die Epidemiologie ist eine Disziplin innerhalb der Gesundheitswissenschaften, die sich mit der Verbreitung von Krankheiten in Bevölkerungen befasst. Im Mittelpunkt stehen die Analyse der Verbreitungsmuster, die Auswirkungen auf die Population sowie die Erforschung der Ursachen von Krankheiten samt möglicher Vorbeugungsmaßnahmen. Epidemiologische Modelle stellen in diesem Kontext eine Methode zur Untersuchung der Verbreitungsdynamik einer Krankheit dar. Ein Anwendungsfall besteht darin, sie zur Abschätzung der Wirksamkeit von Interventionsmaßnahmen bei Eindämmung der Krankheitsausbreitung heranzuziehen.
Eine verbreitete Kategorie epidemiologischer Modelle sind die sogenannten Kompartiment-Modelle, die die Verbreitung von Infektionskrankheiten behandeln. Sie partitionieren die betrachtete Population in verschiedene Gruppen und modellieren den zeitlichen Austausch zwischen diesen Gruppen mithilfe eines Systems von Differentialgleichungen. Der Ursprung dieser Modelle ist auf Anfang des 20. Jahrhunderts datiert und findet sich in Arbeiten von Hamer [5], Ross [17] sowie Kermack und Mc Kendrick [9]. Die beiden folgenden Kapitel stützen sich jedoch vornehmlich auf das Werk *Three Basic Epidemiological Models* von Hethcote [6], während sie mit dem SI- und SIR-Modell zwei der grundlegendsten Kompartimentmodelle einführen. Im anschließenden dritten Kapitel erfolgt darauf aufbauend die Entwicklung eines komplexeren epidemiologischen Modells, das die Zielsetzung verfolgt, die Ausbreitungsdynamik der modellierten Infektionskrankheit vielschichtiger aufzulösen.

3.1 SI-Modell

Das SI-Modell ist das einfachste Kompartimentmodell und bildet die Grundlage für komplexere Modelle, wie beispielsweise das im weiteren Verlauf besprochene SIR-Modell. Namensgebend für das Modell ist die Unterteilung der Bevölkerung in die beiden disjunkten Klassen der s̲uszeptiblen und der i̲nfizierten Individuen. Im nachfolgenden Abschnitt zu den Modellannahmen wird erläutert, auf welchen Voraussetzungen das Modell beruht.

3.1.1 Modellannahmen

3.1.1.1 Population

Es wird davon ausgegangen, dass die Größe N der betrachteten Population über die Zeit hinweg konstant bleibt:

$$N = S(t)N + I(t)N$$

Hierbei bezeichnet $S(t)$ den Populationsanteil der suszeptiblen Individuen und $I(t)$ entsprechend den der infizierten Individuen. Eine Erweiterung des Modells um Geburten und Sterbefälle ist möglich. In diesem Fall wird angenommen, dass pro Tag gleich viele Geburten wie Todesfälle auftreten, was die Populationsgröße konstant hält. Die modellierten Todesfälle sind in diesem Zusammenhang von natürlicher Art und resultieren nicht aus der Krankheit selbst. Vielmehr treten die Todesfälle proportional zur Populationsgröße mit Proportionalitätskonstante μ, die als tägliche *Sterberate* (beziehungsweise *Geburtenrate*) bezeichnet wird, auf. Die Altersstruktur entspricht damit einer Exponentialverteilung mit einer durchschnittlichen Lebenserwartung von $\frac{1}{\mu}$ Tagen.

3.1.1.2 Übertragung

Im Modell wird von einer direkten Übertragung der Krankheit von Mensch zu Mensch ausgegangen. Dabei wird angenommen, dass alle Individuen direkt nach ihrer Ansteckung in gleicher Intensität infektiös sind und die Population einer uniformen Durchmischung unterliegt, was zur Folge hat, dass die Kontakte zwischen den Individuen rein zufällig erfolgen. Die tägliche Kontaktanzahl eines Individuums wird mit κ bezeichnet. Gemeinsam mit der Wahrscheinlichkeit β einer Infektionsübertragung beim Zusammentreffen eines suszeptiblen und eines infektiösen Individuums bildet sie die *tägliche Kontaktrate* $\lambda := \kappa \cdot \beta$. Diese zeigt an, wie viele adäquate Kontakte ein Individuum pro Tag hat. Ein adäquater Kontakt zwischen zwei Individuen ist gekennzeichnet durch die Übertragung der Krankheit, falls ein suszeptibles und ein infektiöses Individuum beteiligt sind. Daraus resultiert, dass ein Infizierter pro Tag im Durchschnitt λS Suszeptible ansteckt, da S als die Wahrscheinlichkeit verstanden werden kann, mit der es sich bei einem zufälligen Individuum um ein suszeptibles Individuum handelt. Die *Inzidenz* zum Zeitpunkt t ist die Anzahl aller neu infizierten Suszeptiblen und ist demzufolge durch $\lambda S(t)NI(t)$ gegeben. Die Gesamtzahl aller erkrankten Individuen ist durch $NI(t)$ beschrieben und wird als *Prävalenz* bezeichnet.

3.1.1.3 Keine Immunität

Eine weitere wichtige Modellannahme besteht darin, dass die Erkrankung nicht zur Immunitätsbildung führt und keine Todesfälle verursacht. Das impliziert, dass ein Infizierter nach

seiner Genesung wieder in das Kompartiment der Suszeptiblen zurückkehrt. Die tägliche Rate, mit der dieser Übergang erfolgt, wird als *tägliche Genesungsrate* bezeichnet und mit γ notiert. Pro Tag wechseln $\gamma NI(t)$ Infizierte zurück in das Kompartiment der Suszeptiblen. Hierdurch ergibt sich eine durchschnittliche Infektionsdauer von $\frac{1}{\gamma}$ Tagen. Unter Einschluss der Sterberate erhält man die angepasste durchschnittliche Infektionsdauer $\frac{1}{\mu+\gamma}$, die im Produkt mit der täglichen Kontaktrate die *Kontaktzahl* $\sigma := \frac{\lambda}{\mu+\gamma}$ ergibt. Die Kontaktzahl gibt an, wie viele Ansteckungen durch ein erstes infiziertes Individuum während seiner gesamten Infektionsdauer in einer ansonsten ausschließlich suszeptiblen Bevölkerung hervorgerufen werden. Die *Reproduktionszahl* zum Zeitpunkt t ist hingegen durch $\sigma S(t)$ gegeben und beschreibt, wie viele Ansteckungen ein Infizierter zum Zeitpunkt t im Verlauf seiner gesamten Infektion durchschnittlich verursacht.

3.1.2 Anfangswertproblem

Wie weiter oben bereits erwähnt, wird der Austausch zwischen den Kompartimenten durch ein System von Differentialgleichungen beschrieben. Die zugehörigen gewöhnlichen Differentialgleichungen sind von erster Ordnung und konstituieren im Verbund mit Anfangswerten für I und S ein Anfangswertproblem:

$$\begin{aligned}
&\frac{d}{dt}NS(t) = -\lambda S(t)NI(t) + \gamma NI(t) + \mu N(t) - \mu NS(t)\\
&\frac{d}{dt}NI(t) = \lambda S(t)NI(t) - \gamma NI(t) - \mu NI(t)\\
&NS(0) = NS_0 > 0, NI(0) = NI_0 > 0 \quad \text{mit } NS_0 + NI_0 = N
\end{aligned} \tag{3.1}$$

Teilt man diese Differentialgleichungen und Anfangswerte durch die konstante Populationsgröße N, erhält man ein Anfangswertproblem, dass das SI-Modell in Bezug auf die Kompartimentanteile beschreibt.

$$\begin{aligned}
&\frac{d}{dt}S(t) = -\lambda S(t)I(t) + \gamma I(t) + \mu(t) - \mu S(t)\\
&\frac{d}{dt}I(t) = \lambda S(t)I(t) - \gamma I(t) - \mu I(t)\\
&S(0) = S_0 > 0, I(0) = I_0 > 0 \quad \text{mit } S_0 + I_0 = 1
\end{aligned} \tag{3.2}$$

Genau wie für etwaige Lösungen, werden auch für alle Parameter (λ, γ und μ) negative Varianten ausgeschlossen, da diese ohne epidemiologisch sinnvolle Entsprechungen in der realen Welt sind. Eine anschauliche Darstellung des Austauschs zwischen den beiden Kompartimenten des SI-Modells findet sich in Abbildung 3.1.

Anmerkung 3.1
Dass die Population im zeitlichen Verlauf bei konstanter Größe bleibt, folgt aus

$$\frac{d}{dt}S(t) + \frac{d}{dt}I(t) = 0$$

bzw.

$$\frac{d}{dt}NS(t) + \frac{d}{dt}NI(t) = 0.$$

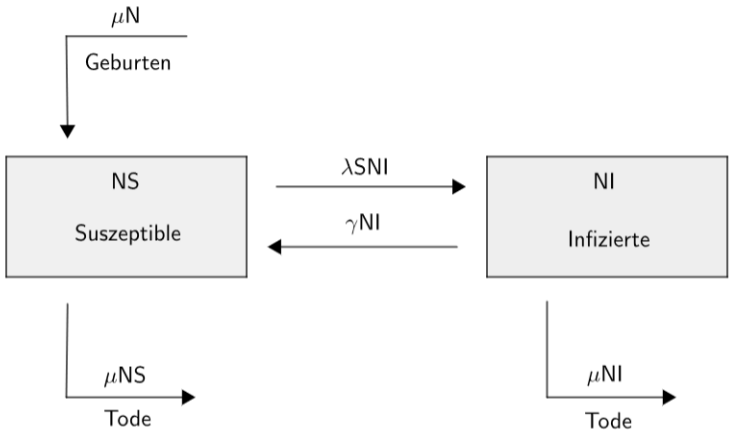


Abbildung 3.1: Kompartimentdiagramm des SI-Modells unter Einbezug von Geburten und Todesfällen.

3.1.3 Analyse

Die vorliegende Analyse bezieht sich auf das Anfangswertproblem der Kompartimentanteile (3.2), lässt sich aber geradewegs auf die absolute Variante übersetzen. Zu jedem Zeitpunkt kann der Anteil an Suszeptiblen über den Anteil der Infizierten gemäß

$$S(t) = 1 - I(t)$$

ausgedrückt werden. Daher genügt es, das Anfangswertproblem

$$\begin{aligned} &\frac{d}{dt}I(t) = (\lambda - (\gamma + \mu))\, I(t) - \lambda I(t)^2 \\ &I(0) = I_0 > 0 \end{aligned} \tag{3.3}$$

zu betrachten.

Theorem 3.2
Die Lösung von Anfangswertproblem (3.3) ist

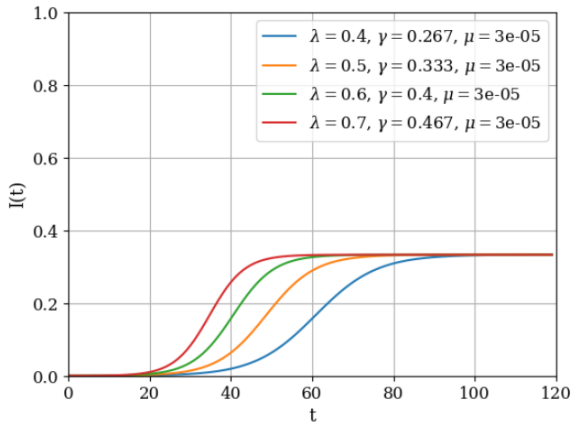
$$I(t) = \begin{cases} \frac{e^{(\gamma+\mu)(\sigma-1)t}}{\frac{\sigma}{\sigma-1}(e^{(\gamma+\mu)(\sigma-1)t}-1)+\frac{1}{I_0}} & \sigma \neq 1 \\ \frac{1}{\lambda t+\frac{1}{I_0}} & \sigma = 1 \end{cases},$$

wobei $\sigma = \frac{\lambda}{\gamma+\mu}$ der Kontaktzahl entspricht.

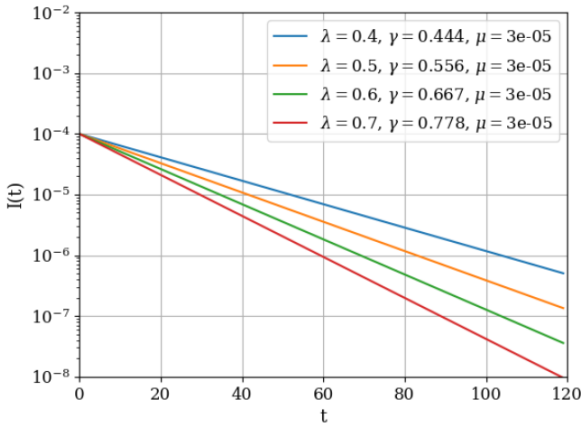
Beweis. Die Lösung findet sich auf Seite 125 in [6]. □

Theorem 3.3
Die Lösung $I(t)$ von Anfangswertproblem (3.3) strebt gegen $1 - \frac{1}{\sigma}$ für $t \to \infty$ falls $\sigma > 1$ beziehungsweise gegen 0 für $t \to \infty$ falls $\sigma \leqslant 1$.

Beweis. Das Theorem entspricht Theorem 4.1 in [6]. □



(a) Wegen $\sigma > 1$ streben die Lösungen vom Anfangswert $I_0 = 1e-4$ gegen das endemische Gleichgewicht bei $1 - \frac{1}{\sigma} \approx 1/3$ und die Krankheit persistiert.



(b) Wegen $\sigma \leqslant 1$ streben die Lösungen mit Anfangswert $I_0 = 1e-4$ gegen null und die Krankheit stirbt aus.

Abbildung 3.2: Lösungen für Anfangswertprobleme des SI-Modells. In Plot **(a)** sind die Parameter λ, γ und μ so gewählt, dass stets $\sigma \approx 3/2$ gilt, während sie in Plot **(b)** so bestimmt sind, dass stets $\sigma \approx 0.9$ gilt.

Das hat zur Folge, dass die Kontaktzahl σ einem Schwellenwert-Parameter entspricht. Überschreitet die Kontaktzahl einer Krankheit den Schwellenwert eins, so strebt die Prävalenz der Krankheit in der Population gegen ein endemisches Gleichgewicht, während ein Unterschreiten von eins zum Aussterben der Krankheit führt (siehe Abbildung 3.2). Das endemische Gleichgewicht tritt ein, sobald ein Infizierter während seiner gesamten Infektionsdauer im Durchschnitt genau einen Suszeptiblen ansteckt. Dies ist genau dann der Fall,

wenn die Reproduktionszahl σS eins ergibt. Infolgedessen stimmt die Prävalenz im endemischen Gleichgewicht mit dem Produkt aus Inzidenz und durchschnittlicher Infektionsdauer überein, wie die nachstehende Rechnung zeigt:

$$\lambda SNI\frac{1}{\gamma+\mu} = \frac{\lambda}{\gamma+\mu}SNI = \underbrace{\sigma S}_{=1}NI = NI$$

3.2 SIR-Modell

Das SIR-Modell ergänzt das SI-Modell um die Klasse $R(t)$ (von englisch „recovered“ beziehungsweise „removed“), deren Individuen Immunität gegenüber der modellierten Krankheit aufweisen. Das SIR-Modell eignet sich vor allem zum Modellieren von Epidemien. Eine Epidemie bezeichnet die überdurchschnittliche Zunahme von Krankheitsfällen in einem zeitlich und örtlich begrenzen Rahmen. Klassische Beispiele für Krankheiten, die epidemisch auftreten sind Grippe, Windpocken sowie neuerdings Covid-19. Die Annahmen des SIR-Modells stimmen weitestgehend mit denen des SI-Modells überein und werden im Folgenden zusammengefasst.

3.2.1 Modellannahmen

3.2.1.1 Population

Erneut weist die zugrundeliegende Bevölkerung eine im zeitlichen Verlauf konstante Größe N auf, die sich auf die drei Kompartimente der suszeptiblen, infizierten und immunen Individuen aufteilt:

$$N = S(t)N + I(t)N + R(t)N.$$

Dabei bezeichnen $S(t)$, $I(t)$ und $R(t)$ den jeweiligen Anteil des zugehörigen Kompartiments an der Gesamtbevölkerung. Es besteht die Option, in das Modell Geburten und Sterbefälle zu inkludieren. Wird diese Option gewählt, so wird genau wie beim SI-Modell die Annahme getroffen, dass die tägliche Sterberate μ mit der täglichen Geburtenrate übereinstimmt, sodass die Bevölkerungsgröße konstant bleibt.

3.2.1.2 Übertragung

Die Annahmen zur Übertragung der Krankheit sind dieselben wie beim SI-Modell (siehe Unterabschnitt 3.1.1.2), weswegen die Notation im Sinne der täglichen Kontaktrate λ übernommen wird.

3.2.1.3 Immunität

Anders als im SI-Modell erwerben Individuen im SIR-Modell durch eine überstandene Infektion dauerhaft Immunität gegenüber der Krankheit. Modelle, die neben infektionsinduzierter Immunität auch Immunität durch Impfung berücksichtigen, werden als SIRV-Modelle bezeichnet [18][14]. Nach überstandener Infektion wandert ein Individuum in das Kompartiment der Immunen. Die tägliche Rate, mit der dies geschieht, wird wie gehabt tägliche Genesungsrate genannt und mit γ notiert. Hierdurch ergibt sich eine durchschnittliche Infektionsdauer von $\frac{1}{\gamma+\mu}$ Tagen. Damit können die Kontaktzahl und die Reproduktionszahl analog zum SI-Modell durch $\sigma := \frac{\lambda}{\gamma+\mu}$ beziehungsweise $\sigma S(t)$ definiert werden.

3.2.2 Anfangswertproblem

Für die folgenden Betrachtungen wird die Variante des SIR-Modells herangezogen, bei der Geburten und Sterbefälle keine Berücksichtigung finden. Dies hat in der Regel einen vernachlässigbaren Effekt, da wie weiter oben bereits erwähnt, das SIR-Modell sich vornehmlich zur Modellierung kurzer Zeiträume eignet. Der Ausschluss von Geburten und Todesfällen ist gleichzusetzen mit einer Sterbe- beziehungsweise Geburtenrate von $\mu = 0$. Konsequenterweise ist die Kontaktzahl in diesem Fall durch $\sigma := \frac{\lambda}{\gamma}$ gegeben. Das Anfangswertproblem des SIR-Modells mit absoluten Kompartimentgrößen ist somit gegeben durch:

$$\begin{aligned}
&\frac{d}{dt}NS(t) = -\lambda S(t)NI(t)\\
&\frac{d}{dt}NI(t) = \lambda S(t)NI(t) - \gamma NI(t)\\
&\frac{d}{dt}NR(t) = \gamma NI(t)\\
&NS(0) = NS_0 > 0, NI(0) = NI_0 > 0, NR(0) = NR_0 \geqslant 0\\
&\text{mit } NS_0 + NI_0 + NR_0 = N
\end{aligned} \tag{3.4}$$

Teilen durch die Populationsgröße N liefert das Anfangswertproblem mit relativen Kompartimentgrößen:

$$\begin{aligned}
&\frac{d}{dt}S(t) = -\lambda S(t)I(t)\\
&\frac{d}{dt}I(t) = \lambda S(t)I(t) - \gamma I(t)\\
&\frac{d}{dt}R(t) = \gamma I(t)\\
&S(0) = S_0 > 0, I(0) = I_0 > 0, R(0) = R_0 \geqslant 0\\
&\text{mit } S_0 + I_0 + R_0 = 1
\end{aligned} \tag{3.5}$$

Genau wie für etwaige Lösungen, werden auch für die Parameter λ und γ negative Varianten ausgeschlossen, da keine epidemiologisch sinnvolle Übersetzung in die reale Welt möglich wäre. Der in Anfangswertproblem (3.4) definierte Austausch zwischen den Kompartimenten wird in Abbildung 3.3 zusammengefasst.

Anmerkung 3.4
Da für beide Varianten des Anfangswertproblems

$$\frac{d}{dt}S(t) + \frac{d}{dt}I(t) + \frac{d}{dt}R(t) = 0$$

bzw.

$$\frac{d}{dt}NS(t) + \frac{d}{dt}NI(t) + \frac{d}{dt}NR(t) = 0$$

gilt, bleibt die Populationsgröße im zeitlichen Verlauf konstant.

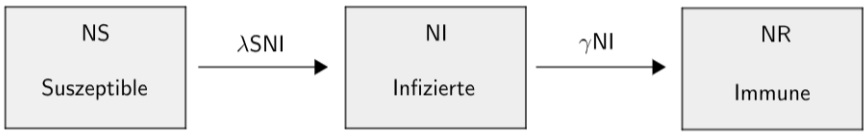


Abbildung 3.3: Kompartimentdiagramm des SIR-Modells unter Ausschluss von Geburten und Sterbefällen.

3.2.3 Analyse

Zunächst wird festgehalten, dass sich der Anteil an Immunen zu jedem Zeitpunkt durch die Anteile der beiden anderen Kompartimente mittels

$$R(t) = 1 - S(t) - I(t)$$

ausdrücken lässt. Demzufolge genügt es, Anfangswertproblem (3.5) in der SI-Phasenebene zu betrachten. Da, wie zuvor erwähnt, negative Lösungen ausgeschlossen werden, wird sich hierbei auf folgendes Gebiet beschränkt:

$$T := \{(S, I) \mid S \geqslant 0, I \geqslant 0, S + I \leqslant 1\}$$

Wie das anschließende Theorem zeigt, kommt die anfängliche Reproduktionszahl einem Schwellenwert-Parameter gleich, da sich an ihr entscheidet, ob die zugehörige Krankheit einen epidemischen Verlauf nimmt oder ohne ein vorübergehendes Ansteigen der Prävalenz direkt wieder ausstirbt.

Theorem 3.5
Sei $(S(t), I(t))$ eine Lösung von Anfangswertproblem (3.5). Gilt $\sigma S_0 \leqslant 1$, so strebt $I(t)$ gegen null für $t \to \infty$. Gilt hingegen $\sigma S_0 > 1$, so wächst $I(t)$ zunächst bis zu einem Maximalwert $I_{\max}$ an und strebt erst anschließend gegen null für $t \to \infty$. Der Maximalwert ist durch

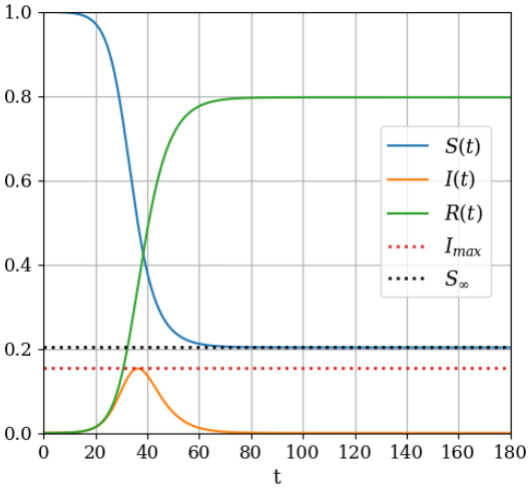
$$I_{\max} := 1 - R_0 - \frac{1}{\sigma} - \frac{\ln(\sigma S_0)}{\sigma}$$

gegeben. Ferner ist der Anteil des suszeptiblen Kompartiments $S(t)$ eine beschränkte monoton fallende Funktion. Der zugehörige Grenzwert S_∞ entspricht der eindeutigen Lösung aus dem Intervall $(1, \frac{1}{\sigma})$ der Gleichung

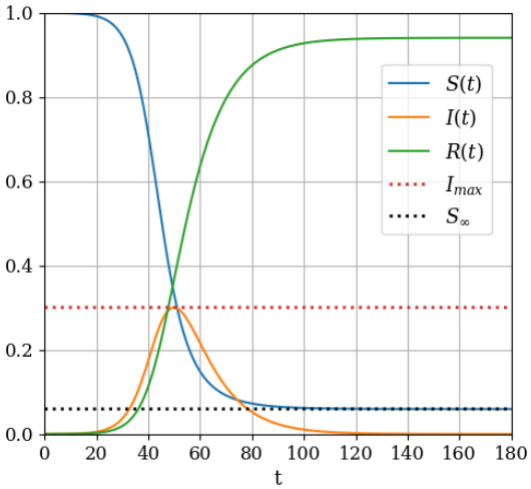
$$1 - R_0 - S_\infty + \frac{\ln\left(\frac{S_\infty}{S_0}\right)}{\sigma} = 0.$$

Beweis. Das Theorem entspricht Theorem 5.1 in [6] □

Im Falle einer Epidemie geht der Prävalenzanstieg in ein Abfallen der Prävalenz über, sobald ein Infizierter im Durchschnitt nur noch weniger als einen Suszeptiblen ansteckt. Dies ist gegeben, wenn $S(t) < \frac{1}{\sigma}$ gilt, da die Reproduktionszahl $\sigma S(t)$ dann unter eins liegt. Abbildung 3.4 zeigt epidemische Verläufe für eine anfängliche Reproduktionszahl, die größer als eins ist, während Abbildung 3.5 beispielhafte Verläufe für eine anfängliche Reproduktionszahl, die kleiner als eins ist, darstellt.

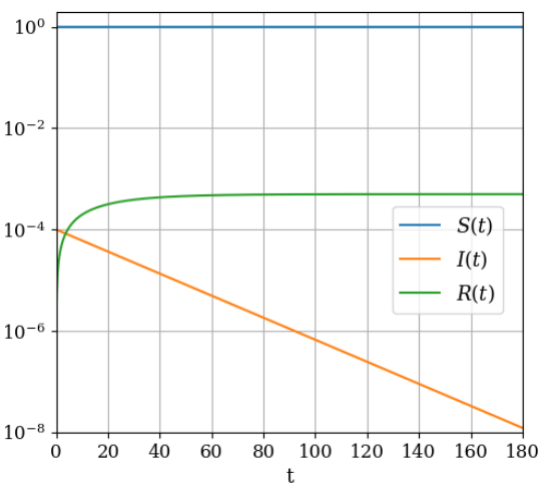


(a) Parameter: $\lambda = 0.5,\ \gamma = 0.25$
$\Rightarrow \sigma = 2$

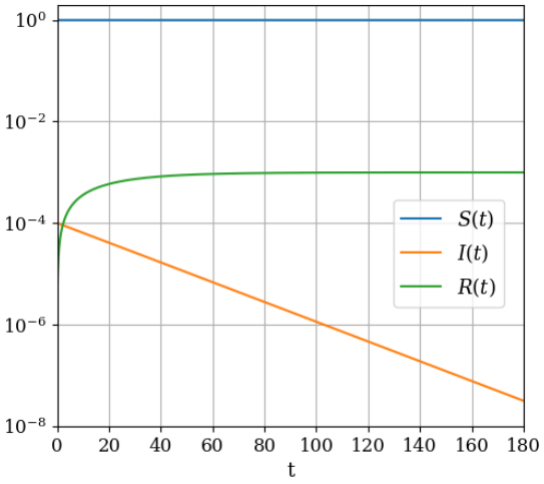


(b) Parameter: $\lambda = 0.3,\ \gamma = 0.1$
$\Rightarrow \sigma = 3$

Abbildung 3.4: Lösungen für Anfangswertprobleme des SIR-Modells mit epidemischem Verlauf bei einer anfänglichen Prävalenz von $I_0 = 1e - 4$.



(a) Parameter: $\lambda = 0.2,\ \gamma = 0.25$ $\Rightarrow \sigma = 0.8$



(b) Parameter: $\lambda = 0.4,\ \gamma = 0.445$ $\Rightarrow \sigma = 0.9$

Abbildung 3.5: Lösungen für Anfangswertprobleme des SIR-Modells mit ausbleibender Epidemie bei einer Startprävalenz von $I_0 = 1e - 4$.

3.3 Erweitertes SIR-Modell

In diesem Kapitel erfolgt die Entwicklung eines epidemiologischen Modells, das die Komplexität des zuvor vorgestellten SIR-Modells durch räumliches Auflösen des Infektionsgeschehens und Einbeziehen von Alter und Blutgruppe der Individuen erhöht. Das Interesse für die räumliche Auflösung liegt darin begründet, dass Epidemien mit Unterschieden in der räumlichen Ausprägung einhergehen, welche das klassische SIR-Modell nicht abzubilden vermag. Die Einbeziehung der Altersstruktur und der Blutgruppenverteilung der Bevölkerung gründet auf der Annahme, dass sich diese auf die Ausbreitungsdynamik der Krankheit auswirken könnten. Als Beispiel sei in diesem Kontext die Beobachtung genannt, dass junge Menschen vornehmlich mit jungen Menschen und alte Menschen hauptsächlich mit alten Menschen Kontakte haben. Auf Grundlage von [10] wird diese Beobachtung in den später durchgeführten Experimenten quantitativ unterlegt. Die Blutgruppen betreffend gab es beispielsweise im Zuge der Covid-19 Pandemie Signale, dass diese auf die ein oder andere Art die Ansteckungswahrscheinlichkeit beeinflussen könnten [21][15][1].

3.3.1 Modellannahmen

In diesem Unterabschnitt wird zunächst die Struktur der betrachteten Bevölkerung hinsichtlich Bevölkerungsdichte, Altersstruktur und Blutgruppenverteilung dargelegt und anschließend die damit verbundenen Annahmen zur Krankheitsübertragung vorgestellt.

3.3.1.1 Population

Die betrachtete Population besiedelt die Fläche $\Omega = [0, X] \times [0, Y] \subset \mathbb{R}^2$, wobei die Bevölkerungsdichte als homogen und zeitlich konstant angenommen wird: $f_\Omega(x, y) = \frac{N}{XY}$. Die demzufolge ebenfalls konstante Bevölkerungsgröße N ist somit durch

$$N = \int_0^X \int_0^Y f_\Omega(x, y)\, dxdy$$

gegeben. Zusätzlich zur räumlichen Auflösung unterteilt sich die Bevölkerung in N_a viele Altersklassen $A = \{a_1, \ldots, a_{N_a}\}$, wobei der Anteil einer Altersklasse a_i an der Gesamtbevölkerung durch die zeit- und ortsunabhängige Funktion $f_A(a_i)$ gegeben ist:

$$N = \sum_{i=1}^{N_a} N \cdot f_A(a_i)$$

Unabhängig von den Altersklassen werden die Individuen der Population in N_b viele Blutgruppen $B = \{b_1, \ldots, b_{N_b}\}$ unterteilt, wobei der Anteil einer Blutgruppe b_i analog zu den Altersklassen durch die zeit- und ortsunabhängige Funktion $f_B(b_i)$ bestimmt ist:

$$N = \sum_{i=1}^{N_b} N \cdot f_B(b_i)$$

Da Unabhängigkeit von Alter, Blutgruppe und Ort vorausgesetzt werden, folgt für die Populationsgröße:

$$\begin{aligned} N &= \int_0^X \int_0^Y \sum_{i=1}^{N_a} \sum_{j=1}^{N_b} f_\Omega(x, y) \cdot f_A(a_i) \cdot f_B(b_j)\, dxdy \\ &= \int_0^X \int_0^Y \sum_{i=1}^{N_a} \sum_{j=1}^{N_b} \underbrace{\frac{N \cdot f_A(a_i) \cdot f_B(b_j)}{XY}}_{=:N(a_i, b_j)} \end{aligned} \tag{3.6}$$

Damit gilt ferner, dass sich für jede Kombination aus Altersklasse und Blutgruppe zu jedem Zeitpunkt und an jedem Ort stets gleich viele Individuen dieser Kombination befinden. Für Altersklasse a_i und Blutgruppe b_j ist dieser Wert durch $N(a_i, b_j)$ gegeben. Ausgehend von der durch $N(a_i, b_j)$ beschriebenen Bevölkerungsstruktur werden nun die disjunkten Kompartimente der suszeptiblen, infizierten und immunen Individuen wie folgt eingeführt:

$$N(a_i, b_j) = S(t, a_i, b_j, x, y) + I(t, a_i, b_j, x, y) + R(t, a_i, b_j, x, y) \tag{3.7}$$

Hierbei sind $S(t, a_i, b_j, x, y)$, $I(t, a_i, b_j, x, y)$ und $R(t, a_i, b_j, x, y)$ die Anzahlen der Suszeptiblen, Infizierten beziehungsweise Immunen am Ort (x, y) zum Zeitpunkt t. Zu verschiedenen Zeit- und Ortspunkten unterscheidet sich damit zwar nicht der Wert von $N(a_i, b_j)$, dafür aber die Zusammensetzung aus suszeptiblen, infizierten und immunen Individuen (siehe Abbildung 3.6). Einsetzen von Gleichung (3.7) in Gleichung (3.6) liefert schließlich eine Darstellung der Populationsgröße auf Grundlage der Gesamtzahlen an Suszeptiblen, Infizierten und Immunen:

$$\begin{aligned} N = &\int_0^X \int_0^Y \sum_{i=1}^{N_a} \sum_{j=1}^{N_b} S(t, a_i, b_j, x, y)\, dxdy \\ &+ \int_0^X \int_0^Y \sum_{i=1}^{N_a} \sum_{j=1}^{N_b} I(t, a_i, b_j, x, y)\, dxdy \\ &+ \int_0^X \int_0^Y \sum_{i=1}^{N_a} \sum_{j=1}^{N_b} R(t, a_i, b_j, x, y)\, dxdy \end{aligned}$$

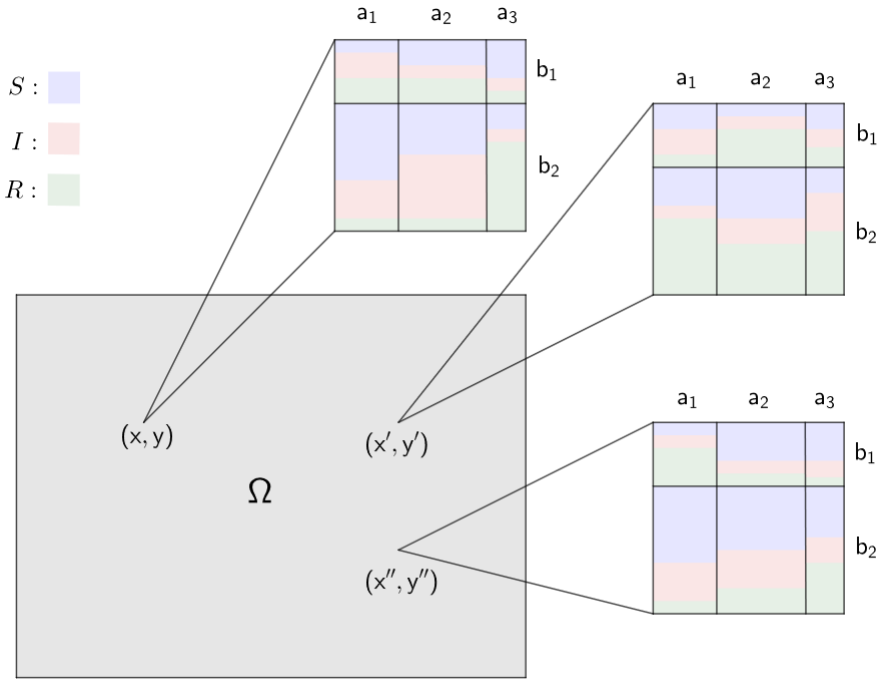


Abbildung 3.6: Schematische Darstellung des Zustandes einer exemplarischen Population mit drei Altersklassen und zwei Blutgruppen. Für drei Ortspunkte werden beispielhaft die Aufteilungen in suszeptible, infizierte und immune Individuen hervorgehoben.

3.3.1.2 Übertragung

Es wird vorausgesetzt, dass die modellierte Krankheit von Mensch zu Mensch übertragen wird und eine Ansteckung ohne Latenzphase zur sofortigen Infektiosität des angesteckten Individuums führt. Die Übertragungen finden in jedem Ortspunkt $(x, y) \in \Omega$ separat statt und basieren auf Kontakten, die als abhängig vom Alter der betroffenen Individuen angenommen werden. Der Parameter $\kappa \in \mathbb{R}^{N_a \times N_a}$, genannt *Kontaktmatrix*, trägt dieser Abhängigkeit Rechnung, indem $\kappa[i, j]$ die tägliche Kontaktanzahl eines Individuums aus Altersklasse a_j mit Individuen aus Altersklasse a_i enthält. Die Wahrscheinlichkeit, dass ein Kontakt zur Ansteckung führt, hängt wiederum von den Blutgruppen der betroffenen Individuen ab und lässt sich an der *Wahrscheinlichkeitsmatrix* $\beta \in \mathbb{R}^{N_b \times N_b}$ ablesen, wobei $\beta[i, j]$ die Wahrscheinlichkeit dafür angibt, dass ein Kontakt zwischen einem suszeptiblen Individuum mit Blutgruppe b_i und einem infektiösen Individuum mit Blutgruppe b_j zu einer Ansteckung führt. Gemeinsam bestimmen κ und β die *skalierte tägliche Kontaktrate* λ wie folgt:

$$\begin{aligned} &\lambda \in \mathbb{R}^{N_a \times N_b \times N_a \times N_b} \\ &\lambda[i, j, k, l] := \kappa[i, k] \cdot \beta[j, l] \cdot f_B(b_j) \cdot \frac{1}{N(a_i, b_j)} \end{aligned} \tag{3.8}$$

Dieser Definition folgend entspricht $S(t, a_i, b_j, x, y) \cdot \lambda[i, j, k, l]$ der durchschnittlichen Anzahl an Individuen aus Altersklasse a_i mit Blutgruppe b_j, die ein Infizierter aus Altersklasse a_k mit Blutgruppe b_l zum Zeitpunkt t am Ort (x, y) ansteckt. Die durchschnittliche Gesamtzahl an Ansteckungen, die dieser Infizierte verursacht, ist demzufolge durch

$$\sum_{i=1}^{N_a} \sum_{j=1}^{N_b} S(t, a_i, b_j, x, y) \lambda[i, j, k, l]$$

gegeben. Gemeinsam verursachen alle $I(t, a_k, b_l, x, y)$ Infizierten somit durchschnittlich

$$\sum_{i=1}^{N_a} \sum_{j=1}^{N_b} S(t, a_i, b_j, x, y) \lambda[i, j, k, l] I(t, a_k, b_l, x, y)$$

neue Ansteckungen. Umgekehrt berechnet sich die durchschnittliche Anzahl aller zum Zeitpunkt t neu angesteckten Suszeptiblen am Ort (x, y) aus Altersklasse a_i und mit Blutgruppe b_j durch

$$S(t, a_i, b_j, x, y) \sum_{k=1}^{N_a} \sum_{l=1}^{N_b} \lambda[i, j, k, l] I(t, a_k, b_l, x, y).$$

3.3.1.3 Räumliche Ausbreitung

Die räumliche Ausbreitung der Krankheit ergibt sich durch ungerichtete Zufallsbewegungen der Individuen auf Ω, die durch Diffusion abgebildet werden. Widergespiegelt wird die Beweglichkeit der Individuen im Diffusionskoeffizienten D, welcher für alle Individuen gleich ist.

3.3.1.4 Immunität

Weil von der Krankheit hervorgerufene Todesfälle ausgeschlossen werden, endet die Infektionsphase jedes Individuums mit dem Wechsel in das Kompartiment der Immunen. Die Rate, mit der dieser Übergang pro Tag stattfindet, wird von der altersabhängigen *täglichen Genesungsrate* $\gamma \in \mathbb{R}^{N_a}$ bestimmt: $\gamma[i]$ ist die tägliche Genesungsrate der Individuen aus Altersklasse a_i, womit Individuen dieser Altersklasse durchschnittlich $\frac{1}{\gamma[i]}$ Tage lang infiziert sind. In Kombination mit der skalierten täglichen Kontaktrate λ ergibt sich die alters- und blutgruppenabhängige *Kontaktzahl* $\sigma \in \mathbb{R}^{N_a \times N_b}$ wie folgt:

$$\sigma[k,l] := \sum_{i=1}^{N_a} \sum_{j=1}^{N_b} \frac{\lambda[i,j,k,l]}{\gamma[k]} \cdot N(a_i, b_j)$$

Damit ist $\sigma[k,l]$ die ortsunabhängige, durchschnittliche Anzahl an Ansteckungen, die von einem ersten Infizierten aus Altersklasse a_k mit Blutgruppe b_l während seiner gesamten Infektionsdauer in einer ansonsten komplett suszeptiblen Bevölkerung ausgehen.

3.3.2 Anfangswertproblem

Auf Grundlage obiger Ausführungen ergibt sich das Anfangswertproblem für das um Altersklassen, Blutgruppen und zwei Raumdimensionen erweiterte SIR-Modell wie folgt:

$$\textbf{Für } 0 < t \leqslant T,\ a_i \in A,\ b_j \in B \text{ und } (x,y) \in \Omega \backslash \partial\Omega: \tag{3.9}$$

$$\begin{aligned}
\partial_t S(t,a_i,b_j,x,y) &= -S(t,a_i,b_j,x,y) \sum_{k=1}^{N_a} \sum_{l=1}^{N_b} \lambda[i,j,k,l] I(t,a_k,b_l,x,y) \\
&\quad + D\Delta S(t,a_i,b_j,x,y) \\
\partial_t I(t,a_i,b_j,x,y) &= S(t,a_i,b_j,x,y) \sum_{k=1}^{N_a} \sum_{l=1}^{N_b} \lambda[i,j,k,l] I(t,a_k,b_l,x,y) \\
&\quad - \gamma[i] I(t,a_i,b_j,x,y) + D\Delta I(t,a_i,b_j,x,y) \\
\partial_t R(t,a_i,b_j,x,y) &= \gamma[i] I(t,a_i,b_j,x,y) + D\Delta R(t,a_i,b_j,x,y)
\end{aligned}$$

Für $0 < t \leqslant T$, $a_i \in A$, $b_j \in B$ und $(x, y) \in \partial\Omega$:

$$\partial_\nu S(t, a_i, b_j, x, y) = \partial_\nu I(t, a_i, b_j, x, y) = \partial_\nu R(t, a_i, b_j, x, y) = 0$$

Für $a_i \in A$, $b_j \in B$ und $(x, y) \in \Omega$:

$$S(0, a_i, b_j, x, y) \geqslant 0,\ I(0, a_i, b_j, x, y) \geqslant 0,\ R(0, a_i, b_j, x, y) \geqslant 0$$
$$\text{mit } S(0, a_i, b_j, x, y) + I(0, a_i, b_j, x, y) + R(0, a_i, b_j, x, y) = N(a_i, b_j)$$

Wobei ν das äußere Normalenfeld zu Ω ist. Die Parameter λ, γ und D werden genau wie eventuelle Lösungen auf nicht-negative Werte eingeschränkt, da für die negativen Varianten eine epidemiologisch sinnvolle Interpretation nicht möglich wäre. Die definierte Neumann-Randbedingung trägt dafür Sorge, dass die Individuen Ω nicht verlassen können. Ferner gilt

$$\partial_t S(t, a_i, b_j, x, y) + \partial_t I(t, a_i, b_j, x, y) + \partial_t R(t, a_i, b_j, x, y) = 0, \tag{3.10}$$

da sich einerseits die Reaktionsterme und andererseits die Diffusionsterme gegenseitig aufheben. Dies ist für die Reaktionsterme offensichtlich und folgt für die Diffusionsterme aus nachstehender Rechnung:

$$\begin{aligned}
&D\Delta S(t, a_i, b_j, x, y) + D\Delta I(t, a_i, b_j, x, y) + D\Delta R(t, a_i, b_j, x, y) \\
&= D\Delta(\underbrace{S(t, a_i, b_j, x, y) + I(t, a_i, b_j, x, y) + R(t, a_i, b_j, x, y)}_{=N(a_i, b_j),\ \text{konstant in } (x,y)}) \\
&= 0
\end{aligned}$$

Wegen Gleichung (3.10) ist somit sichergestellt, dass sich die Anzahl an Individuen jeder Kombination aus Altersklasse und Blutgruppe, jeweils gegeben durch $N(a_i, b_j)$, an keinem Ort ändert. Abbildung (3.7) illustriert die lokale Wechselwirkung (den Reaktionsanteil obiger Differentialgleichungen) zwischen den Kompartimenten.

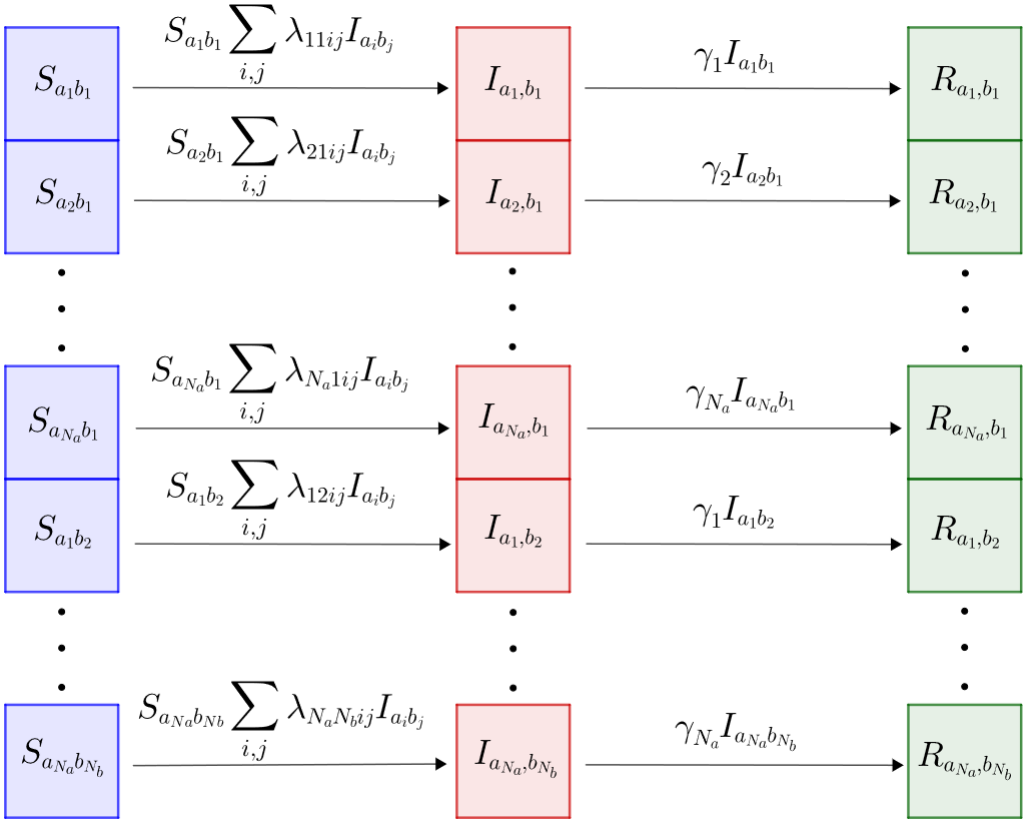


Abbildung 3.7: Kompartimentdiagramm auf Grundlage des erweiterten SIR-Modells. Um eine kompaktere Darstellung zu ermöglichen, wird eine Schreibweise genutzt, die $S(t, a_i, b_j, x, y)$ durch $S_{a_ib_j}$ abgekürzt.

4 Numerische Lösungsverfahren

In diesem Teil der Arbeit werden zwei Lösungsverfahren für Anfangswertproblem (3.9) präsentiert, wobei das erste mit vollen Tensoren arbeitet und das zweite die Dynamik des Modells im hierarchischen Tuckerformat (siehe Kapitel 2.2) approximiert. Um den Einfluss des hierarchischen Tuckerformats besser herausarbeiten zu können, basieren beide Lösungsmethoden auf dem expliziten Eulerverfahren. Es sei weiter hervorgehoben, dass das Kompartiment der Immunen nicht explizit berechnet wird, da es sich entsprechend Gleichung (3.7) jederzeit aus den Kompartimenten der Suszeptiblen und Infizierten bestimmen lässt.

4.1 Explizites Eulerverfahren mit vollen Tensoren

Da Altersklassen und Blutgruppen bereits von vornherein diskret gewählt sind, müssen als erster Schritt nur die Zeit und der Raum diskretisiert werden.

4.1.1 Diskretisierung von Zeit und Raum

Das betrachtete Zeitintervall $[0, T]$ wird in N_t+1 gleichmäßig verteilte Zeitpunkte aufgeteilt:

$$t_n = n\Delta t, \quad n = 0, \dots, N_t, \quad \Delta t = \frac{T}{N_t}$$

Analog wird $\Omega = [0, X] \times [0, Y]$ in $(N_x + 1) \cdot (N_y + 1)$ gleichmäßig verteilte Gitterpunkte aufgeteilt:

$$(x_i, y_j) = (ih, jh), \quad i = 0, \dots, N_x, \quad j = 0, \dots, N_y, \quad h = \frac{X}{N_x} = \frac{Y}{N_y}$$

4.1.2 Diskretisierung der Ortsableitungen

Basierend auf obiger Raumdiskretisierung werden in diesem Abschnitt die zweiten Ortsableitungen des Laplace Operators

$$\Delta u(x_i, y_j) = \partial_{xx} u(x_i, y_j) + \partial_{yy} u(x_i, y_j)$$

unter Einhaltung der Neumann-Randbedingung durch Differenzenquotienten ersetzt. Im Sinne der Lesbarkeit werden die Differenzenquotienten anhand einer exemplarischen Funk-

tion $u(x_i, y_j)$ samt diskreter Gitterfunktion $u_{i,j} = u(x_i, y_j)$ hergeleitet. Da die definierte Neumann-Randbedingung (siehe Anfangswertproblem (3.9)) in x- und y-Richtung gleich wirkt, genügt es, ein Schema zur Diskretisierung der zweiten Ableitungen nach x herzuleiten und anschließend auf die zweiten Ableitungen nach y zu übertragen.
Sei also $(x_0, y_j) \in \partial\Omega$ ein linker Randpunkt und Entwicklungspunkt folgender Taylorpolynome:

$$\begin{aligned}
u_{0,j} &= u(x_0, y_j) \\
u_{1,j} &= u(x_1, y_j) = u_{0,j} + h\partial_x u(x_0, y_j) + \frac{1}{2}h^2\partial_{xx}2u(x_0, y_j) + \mathcal{O}(h^3) \\
u_{2,j} &= u(x_2, y_j) = u_{0,j} + 2h\partial_x u(x_0, y_j) + 2h^2\partial_{xx}u(x_0, y_j) + \mathcal{O}(h^3)
\end{aligned}$$

Da die ersten Ableitungen wegen der Neumann-Randbedingung null sind, vereinfachen sich die Taylorpolynome zu:

$$\begin{aligned}
u_{0,j} &= u(x_0, y_j) \\
u_{1,j} &= u_{0,j} + \frac{1}{2}h^2\partial_{xx}u(x_0, y_j) + \mathcal{O}(h^3) \\
u_{2,j} &= u_{0,j} + 2h^2\partial_{xx}u(x_0, y_j) + \mathcal{O}(h^3)
\end{aligned}$$

Auflösen nach der zweiten Ableitung liefert schließlich:

$$\partial_{xx}u(x_0, y_j) + \mathcal{O}(h) = \frac{u_{0,j} - 2u_{1,j} + u_{2,j}}{h^2}$$

Analog lässt sich die Darstellung für Punkte $(x_{N_x}, y_j) \in \partial\Omega$ auf dem rechten Rand herleiten:

$$\partial_{xx}u(x_{N_x}, y_j) + \mathcal{O}(h) = \frac{u_{N_x-2,j} - 2u_{N_x-1,j} + u_{N_x,j}}{h^2}$$

Der letzte Fall ist durch einen inneren Punkt $(x_i, y_j) \in \Omega$ gegeben:

$$\begin{aligned}
u_{i,j} &= u(x_i, y_j) \\
u_{i-1,j} &= u(x_{i-1}, y_j) = u_{i,j} - h\partial_x u(x_i, y_j) + \frac{1}{2}h^2\partial_{xx}u(x_i, y_j) + \mathcal{O}(h^3) \\
u_{i+1,j} &= u(x_{i+1}, y_j) = u_{i,j} + h\partial_x u(x_i, y_j) + \frac{1}{2}h^2\partial_{xx}u(x_i, y_j) + \mathcal{O}(h^3)
\end{aligned}$$

Auflösen nach der zweiten Ableitung ergibt:

$$\partial_{xx}u(x_i, y_j) + \mathcal{O}(h) = \frac{u_{i-1,j} - 2u_{i,j} + u_{i+1,j}}{h^2}$$

Überführen der Gleichungen in Matrixschreibweise ermöglicht eine gebündelte Darstellung:

$$A_{h,x} := \frac{1}{h^2}\begin{bmatrix} 1 & -2 & 1 & 0 & & \dots & 0 \\ 1 & -2 & 1 & 0 & & \dots & 0 \\ 0 & 1 & -2 & 1 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & 1 & -2 & 1 & 0 \\ 0 & \dots & & 0 & 1 & -2 & 1 \\ 0 & \dots & & 0 & 1 & -2 & 1 \end{bmatrix} \in \mathbb{R}^{(N_x+1)\times(N_x+1)} \tag{4.1}$$

$$U := \begin{bmatrix} u_{0,0} & u_{0,1} & \dots & u_{0,N_y} \\ u_{1,0} & u_{1,1} & \dots & u_{1,N_y} \\ \vdots & \vdots & \ddots & \vdots \\ u_{N_x,0} & u_{N_x,1} & \dots & u_{N_x,N_y} \end{bmatrix} \in \mathbb{R}^{(N_x+1)\times(N_y+1)}$$

$$A_{h,x} \cdot U = \begin{bmatrix} \partial_{xx}u_{0,0} & \partial_{xx}u_{0,1} & \dots & \partial_{xx}u_{0,N_y} \\ \partial_{xx}u_{1,0} & \partial_{xx}u_{1,1} & \dots & \partial_{xx}u_{1,N_y} \\ \vdots & \vdots & \ddots & \vdots \\ \partial_{xx}u_{N_x,0} & \partial_{xx}u_{N_x,1} & \dots & \partial_{xx}u_{N_x,N_y} \end{bmatrix} + \mathcal{O}(h)$$

Die Übertragung des Schemas auf die y-Richtung liefert die nachstehenden Differenzenquotienten für untere Randpunkte, obere Randpunkte und innere Punkte:

$$\partial_{yy}u(x_i, y_0) + \mathcal{O}(h) = \frac{u_{i,0} - 2u_{i,1} + u_{i,2}}{h^2}$$
$$\partial_{yy}u(x_i, y_{N_y}) + \mathcal{O}(h) = \frac{u_{i,N_y} - 2u_{i,N_y-1} + u_{i,N_y-2}}{h^2}$$
$$\partial_{yy}u(x_i, y_j) + \mathcal{O}(h) = \frac{u_{i,j-1} - 2u_{i,j} + u_{i,j+1}}{h^2}$$

Die zugehörige Matrixschreibweise lautet damit:

$$A_{h,y} := \frac{1}{h^2}\begin{bmatrix} 1 & -2 & 1 & 0 & & \dots & 0 \\ 1 & -2 & 1 & 0 & & \dots & 0 \\ 0 & 1 & -2 & 1 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & 1 & -2 & 1 & 0 \\ 0 & \dots & & 0 & 1 & -2 & 1 \\ 0 & \dots & & 0 & 1 & -2 & 1 \end{bmatrix} \in \mathbb{R}^{(N_y+1)\times(N_y+1)} \tag{4.2}$$

$$U \cdot A_{h,y}^T = \begin{bmatrix} \partial_{yy}u_{0,0} & \partial_{yy}u_{0,1} & \dots & \partial_{yy}u_{0,N_y} \\ \partial_{yy}u_{1,0} & \partial_{yy}u_{1,1} & \dots & \partial_{yy}u_{1,N_y} \\ \vdots & \vdots & \ddots & \vdots \\ \partial_{yy}u_{N_x,0} & \partial_{yy}u_{N_x,1} & \dots & \partial_{yy}u_{N_x,N_y} \end{bmatrix} + \mathcal{O}(h)$$

Damit lässt sich der diskrete Laplace Operator Δ_h durch punktweise Addition der oben hergeleiteten Differenzenquotienten bilden:

$$\begin{aligned}
&\bullet\ \Delta_h u_{i,j} = \frac{u_{i-1,j} + u_{i,j-1} - 4ui, j + u_{i+1,j} + u_{i,j+1}}{h^2} \\
&\bullet\ \Delta_h u_{i,0} = \frac{u_{i-1,0} - u_{i,0} + u_{i+1,0} - 2u_{i,1} + u_{i,2}}{h^2} \\
&\bullet\ \Delta_h u_{i,N_y} = \frac{u_{i,N_y-2} + u_{i-1,N_y} - 2u_{i,N_y-1} - u_{i,N_y} + u_{i+1,N_y}}{h^2} \\
&\bullet\ \Delta_h u_{0,j} = \frac{u_{0,j-1} - u_{0,j} - 2u_{1,j} + u_{0,j+1} + u_{2,j}}{h^2} \\
&\bullet\ \Delta_h u_{N_x,j} = \frac{u_{N_x-2,j} - 2u_{N_x-1,j} + u_{N_x,j-1} - u_{N_x,j} + u_{N_x,j+1}}{h^2} \\
&\bullet\ \Delta_h u_{0,0} = \frac{2u_{0,0} - 2u_{1,0} - 2u_{0,1} + u_{2,0} + u_{0,2}}{h^2} \\
&\bullet\ \Delta_h u_{N_x,N_y} = \frac{u_{N_x-2,N_y} + u_{N_x,N_y-2} - 2u_{N_x-1,N_y} - 2u_{N_x,N_y-1} + 2u_{N_x,N_y}}{h^2} + \\
&\bullet\ \Delta_h u_{0,N_y} = \frac{u_{0,N_y-2} - 2u_{0,Ny-1} + 2u_{0,N_y} - 2u_{1,N_y} + u_{2,N_y}}{h^2} \\
&\bullet\ \Delta_h u_{N_x,0} = \frac{u_{N_x-2,0} - 2u_{N_x-1,0} + 2u_{Nx,0} - 2u_{N_x,1} + u_{N_x,2}}{h^2}
\end{aligned} \tag{4.3}$$

Deutlich übersichtlicher ist die Matrixdarstellung:

$$\Delta_h U = A_{h,x}U + UA_{h,y}^T \tag{4.4}$$

4.1.3 Punktweises explizites Eulerverfahren

Obige Diskretisierungen führen in Kombination mit einer Diskretisierung der zeitlichen Ableitung gemäß des expliziten Eulerverfahrens schließlich zu folgendem Schema

$$\begin{aligned}
S^{n+1}[i,j,k,l] = S^n[i,j,k,l] \ + \ \Delta t \ (&-S^n[i,j,k,l] \sum_{m=1}^{N_a}\sum_{p=1}^{N_b} \lambda[i,j,m,p] I^n[m,p,k,l] \\
&+ \ D\Delta_h S^n[i,j,k,l]) \\
I^{n+1}[i,j,k,l] = I^n[i,j,k,l] \ + \ \Delta t \ (&S^n[i,j,k,l] \sum_{m=1}^{N_a}\sum_{p=1}^{N_b} \lambda[i,j,m,p] I^n[m,p,k,l] \\
&- \gamma[i] I^n[i,j,k,l] \ + \ D\Delta_h I^n[i,j,k,l]) \\
S^0[i,j,k,l] = S(0,a_i,b_j,x_k,y_l), \ I^0[i,j,k,l] &= I(0,a_i,b_j,x_k,y_l),
\end{aligned} \tag{4.5}$$

wobei $S^n, I^n \in \mathbb{R}^{N_a \times N_b \times (N_x+1) \times (N_y+1)}$ zwei reelle Tensoren vierter Ordnung sind, die für $n = 0, \ldots, N_t$ die Lösung gemäß

$$(S^n[i,j,k,l], I^n[i,j,k,l]) \approx (S(t_n, a_i, b_j, x_k, y_l), I(t_n, a_i, b_j, x_k, y_l))$$

approximieren.

4.1.4 Tensorwertiges explizites Eulerverfahren

Die tensorwertige Formulierung des punktweisen Verfahrens (4.5) ist durch

$$\begin{aligned} S^{n+1} &= S^n + \Delta t \left(-S^n \star < \lambda, I^n >_{3,4}^{1,2} + D\Delta_h S^n \right) \\ I^{n+1} &= I^n + \Delta t \left(S^n \star < \lambda, I^n >_{3,4}^{1,2} - \gamma \star_1 I^n + D\Delta_h I^n \right) \end{aligned} \tag{4.6}$$

gegeben, wobei $D\Delta_h S^n$ und $D\Delta_h I^n$ entsprechend (4.4) durch

$$\begin{aligned} &D\Delta_h S^n = D(A_x \circ_3 S^n + A_y \circ_4 S^n) \\ &\text{bzw.} \\ &D\Delta_h I^n = D(A_x \circ_3 I^n + A_y \circ_4 I^n) \end{aligned}$$

zu berechnen sind. Die Matrizen A_x und A_y sind dabei wie in (4.1) beziehungsweise (4.2) gegeben. Die zu den Tensoroperationen gehörigen Definitionen finden sich in Abschnitt 2.1.3.

4.2 Rangadaptives Eulerverfahren mit hierarchischen Tuckertensoren

Nachdem das auf vollen Tensoren basierende Verfahren vorgestellt wurde, wird nun auf Basis der Ergebnisse in [16] das auf dem hierarchischen Tuckerformat basierende Verfahren entwickelt. Zur Formulierung dessen erfolgt zuerst die Einführung eines Kürzungsoperators, der die hierarchische Tuckerkürzung (siehe Definition 2.41) samt orthogonaler Projektionen wie in Definition 2.40 zur Grundlage hat.

Definition 4.1 (Kürzungsoperator)
Seien $X \in \mathbb{R}^I$ ein reeller Tensor, T_I der zur Indexmenge I gehörige kanonische Dimensionsbaum mit Tiefe p und $k = (k_t)_{t \in T_I}$ eine Rangverteilung auf T_I. Dann ist der Kürzungsoperator $\mathfrak{T}_k$ wie folgt definiert:

$$\begin{aligned} &\mathfrak{T}_k : \mathbb{R}^I \to \mathcal{H}\text{-Tucker}((k_t)_{t \in T_I}) \\ &X \mapsto \mathfrak{T}_k(X) \end{aligned}$$

$$\mathfrak{T}_k(X) := \prod_{t \in T_I^p} \pi_t ... \prod_{t \in T_I^1} \pi_t X$$

Dabei handelt es sich bei den auftretenden π_t um orthogonale Projektionen der Form $\pi_t = U_t U_t^T$, wobei die Spalten von U_t den k_t dominanten linken Singulärvektoren von $\mathcal{M}_t(X)$ entsprechen.

Dieser Kürzungsoperator unterscheidet sich von der hierarchischen Tuckerkürzung nur durch den Umstand, dass er den hierarchischen Rang $k = (k_t)_{t \in T_I}$, auf den gekürzt wird, in der Notation sichtbar macht. Auf algorithmischer Seite wird $\mathfrak{T}_k$ sowohl von Algorithmus 1 als auch von Algorithmus 3 implementiert. Der Unterschied zwischen beiden Algorithmen liegt darin, dass Algorithmus 1 den zu kürzenden Tensor X in expliziter Form übergeben bekommt, während Algorithmus 2 X im hierarchischen Tuckerformat erhält. Davon abgesehen findet sich $\mathfrak{T}_k$ auch im Verfahren zur Berechnung des elementweisen Produkts zweier hierarchischer Tuckertensoren wieder (siehe Algorithmus 4), da dort das resultierende elementweise Produkt bereits im Zuge seiner Berechnung gekürzt wird, um ein zu starkes Anwachsen des hierarchischen Ranges zu vermeiden.

4.2.1 Allgemeine Einführung samt Konvergenzkriterium

Das verwendete Verfahren wird in diesem Teil allgemeingültig und mit Konvergenzkriterium eingeführt. Erst im Anschluss folgt die Anwendung auf das Anfangswertproblem des erweiterten SIR-Modells. Zunächst wird daher folgendes allgemeines Anfangswertproblem

$$\partial_t f(x,t) = \mathcal{N}(f(x,t),x), \quad f(x,0) = f_0(x) \tag{4.7}$$

betrachtet, wobei es sich bei $f : \Omega \times [0,T] \to \mathbb{R}$ um ein d-dimensionales Skalarfeld auf dem Gebiet $\Omega \subseteq \mathcal{R}^d$ ($d \geqslant 2$) handelt und $\mathcal{N}$ ein nichtlinearer Operator mit eventuell vorhandenen Randbedingungen ist. Diskretisieren von (4.7) in Ω (zum Beispiel mit finiten Differenzen) führt zum nachstehenden System gewöhnlicher Differentialgleichungen:

$$\frac{d\boldsymbol{f}(t)}{dt} = \boldsymbol{N}(\boldsymbol{f}(t)), \quad \boldsymbol{f}(0) = \boldsymbol{f}_0 \tag{4.8}$$

Diskretisieren der Zeit mit konstanter Zeitschrittweite Δt und Ersetzen der Zeitableitung durch Differenzenquotienten im Sinne des klassischen expliziten Eulerverfahrens ergibt

$$u_{n+1} = u_n + \Delta t \boldsymbol{N}(u_n), \tag{4.9}$$

wobei u_n für $n = 0, 1, \ldots$ eine Approximation von $\boldsymbol{f}(n\Delta t)$ darstellt. Anwendung des Kürzungsoperators sowohl auf die Lösung als auch die diskrete Form des nichtlinearen Operators $\mathcal{N}$ in (4.9) liefert schließlich folgendes Schema, das als *rangadaptives Eulerverfahren*

bezeichnet wird:

$$\boldsymbol{f}_{n+1} = \mathfrak{T}_{k_n}\left(\boldsymbol{f}_n + \Delta t \mathfrak{T}_{r_n}(\boldsymbol{N}(\boldsymbol{f}_n))\right) \tag{4.10}$$

Für $n = 0, 1, \ldots$ entspricht $\boldsymbol{f}_n$ hierbei einer Approximation von $\mathfrak{T}_{k_n}(\boldsymbol{f}(n\Delta t))$.

Theorem 4.2
Der globale Fehler des rangadaptiven Eulerverfahrens (4.10) liegt in $\mathcal{O}(\Delta t)$, falls $\boldsymbol{N}$ lipschitzstetig ist und die hierarchischen Ränge k_n und r_n für $n = 0, 1, \ldots$ so gewählt sind, dass die beiden Ungleichungen

$$\begin{aligned}
&||\boldsymbol{N}(\boldsymbol{f}_n) \; - \; \mathfrak{T}_{r_n}(\boldsymbol{N}(\boldsymbol{f}_n))|| \leqslant M_1 \Delta t =: \epsilon_r \\
&||\boldsymbol{f}_n \; + \; \Delta t \; \mathfrak{T}_{r_n}(\boldsymbol{N}(\boldsymbol{f}_n)) \; - \; \mathfrak{T}_{k_n}(\boldsymbol{f}_n \; + \; \Delta t \; \mathfrak{T}_{r_n}(\boldsymbol{N}(\boldsymbol{f}_n)))|| \leqslant M_2 \Delta t^2 =: \epsilon_k
\end{aligned}$$

mit Konstanten $M_1, M_2 > 0$ erfüllt sind.

Beweis. Ein Beweis dazu findet sich [16] unter Abschnitt 4.2. □

Im Kontext der hierarchischen Tuckerkürzung wurde bereits gezeigt, wie aus einer gegebenen Fehlerschranke ein entsprechender hierarchischer Rang ermittelt werden kann, der bei der Kürzung sicherstellt, dass die Fehlerschranke eingehalten wird (siehe Theorem 2.42 beziehungsweise Anmerkung 2.43). Damit kann jederzeit sichergestellt werden, dass die beiden Fehlerschranken $\epsilon_r = M_1 \Delta t$ und $\epsilon_k = M_2 \Delta t^2$ eingehalten werden. Die Autoren in [16] schlagen vor, zuerst eine Zeitschrittweite Δt zu bestimmen, für die das Verfahren mit vollen Tensoren (in dieser Arbeit also das klassische explizite Eulerverfahren) stabil ist und anschließend M_1 und M_2 ungefähr indirekt proportional zu Δt beziehungsweise Δt^2 zu wählen, um ein Ausarten der einzuhaltenden Fehlerschranken zu vermeiden. Trotz dieser Heuristik zur Bestimmung von M_1 und M_2 soll hervorgehoben werden, dass die Konvergenz des rangadaptiven Eulerverfahrens von Theorem 4.2 für beliebige Wahlen von M_1 und M_2 sichergestellt ist.

4.2.2 Anwendung auf das erweiterte SIR-Modell

Das rangadaptive Eulerverfahren für das tensorwertige Anfangswertproblem des erweiterten SIR-Modells (4.6) ergibt sich als

$$\begin{aligned}
S^{n+1} &= \mathfrak{T}_{k_n}\left(S^n + \Delta t \; \mathfrak{T}_{r_n}\left(-\mathfrak{T}_{s_n}(S* <\lambda, I^n>_{3,4}^{1,2}) + D\Delta_h S^n\right)\right) \\
I^{n+1} &= \mathfrak{T}_{\tilde{k}_n}\left(I^n + \Delta t \; \mathfrak{T}_{\tilde{r}_n}\left(\mathfrak{T}_{s_n}(S* <\lambda, I^n>_{3,4}^{1,2}) - \gamma \star_1 I^n + D\Delta_h I^n\right)\right),
\end{aligned} \tag{4.11}$$

wobei die hierarchischen Ränge $k_n, r_n, s_n, \tilde{k}_n$ und $\tilde{r}_n$ bei gegebenen Konstanten M_1, M_2 und Zeitschrittweite Δt so zu wählen sind, dass die Kürzungsoperatoren die folgenden

Fehlerschranken einhalten:

- $\mathfrak{T}_{k_n}$ und $\mathfrak{T}_{\tilde{k}_n}$: $\epsilon_k = \frac{M_2 \Delta t^2}{2}$
- $\mathfrak{T}_{r_n}$, $\mathfrak{T}_{\tilde{r}_n}$ und $\mathfrak{T}_{s_n}$: $\epsilon_r = \frac{M_1 \Delta t}{4}$

Unter diesen Bedingungen sind die in Theorem 4.2 vorausgesetzten Ungleichungen erfüllt (siehe Anhang B.3), sodass dieses zusichert, dass der globale Fehler von Verfahren (4.11) in $\mathcal{O}(\Delta t)$ liegt.

5 Experimente

In diesem abschließenden Teil der Arbeit werden Experimente mit der Zielsetzung durchgeführt, einerseits die Eigenschaften des erweiterten SIR-Modells (siehe Kapitel 3.3) unter verschiedenen Parametereinstellungen zu beleuchten und andererseits die Leistungsfähigkeit des hierarchischen Tuckerformats zu untersuchen. Hierfür werden verschiedene Szenarien unter Verwendung sowohl des expliziten Eulerverfahrens mit vollen Tensoren als auch des rangadaptiven Eulerverfahrens mit hierarchischen Tuckertensoren berechnet. Das im Zuge dieser Arbeit entwickelte Python Framework zur Durchführung der Experimente findet sich auf GitHub [19]. Es enthält neben einem umfassenden Python Paket, das das hierarchische Tuckerformat mit einer Auswahl an Rechenoperationen implementiert, auch Jupyter-Notebooks, die die Parameterkonfigurationen der verschiedenen Simulationen enthalten und die Möglichkeit bieten, diese erneut zu berechnen.

Anmerkung 5.1
Sämtliche Simulationen dieser Arbeit wurden auf einer Maschine mit einem AMD Ryzen 5 5600H Prozessor und 16GB RAM durchgeführt.

5.1 Reaktion ohne Diffusion

Die Berechnungen dieses Kapitels basieren auf dem erweiterten SIR-Modell bei ausgeschalteter Diffusionskomponente und bilden damit den Reaktionsanteil des Modells an einem festgelegten Raumpunkt ab. Um den Einfluss der Kontaktmatrix κ und der Wahrscheinlichkeitsmatrix β herauszuarbeiten, werden bei jeweils zwei möglichen Belegungen für κ und β insgesamt vier Szenarien gerechnet. Hierbei wird β einmal so gewählt, dass die Blutgruppen die Ansteckungswahrscheinlichkeit beeinflussen, und einmal so, dass sie keinen Einfluss haben. Analog wird κ einmal so gewählt, dass die Kontakte zwischen den Individuen rein zufällig erfolgen, und einmal so, dass zwischen den verschiedenen Altersklassen ein definiertes Kontaktmuster besteht. Bevor eine detaillierte Betrachtung der verschiedenen Szenarien erfolgt, soll zunächst der zugrundeliegende Experimentaufbau hinsichtlich Modell- und Lösungsverfahrenkonfiguration erläutert werden.

5.1.1 Modellkonfiguration

5.1.1.1 Fixierte Bevölkerungsstruktur

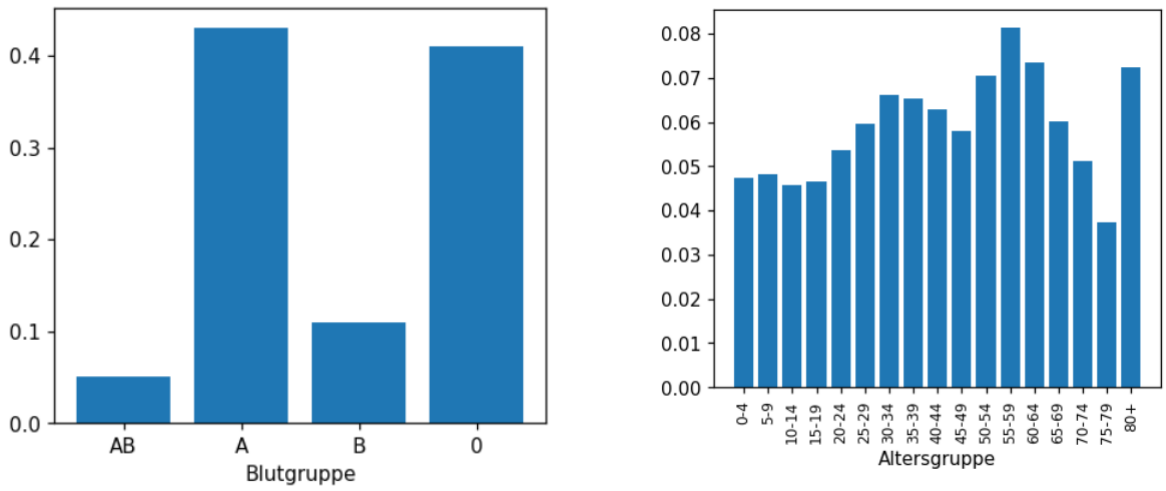
Die Bevölkerungsstruktur wird in allen vier Experimenten mit einer Größe von $N = 10000$ Individuen angenommen, die sich jeweils in eine der 17 Altersklassen

$$\mathcal{A} = \{0 - 4Jahre, 5 - 9Jahre, \ldots, 75 - 79Jahre, 80 + Jahre\}$$

und eine der vier Blutgruppen

$$\mathcal{B} = \{AB, A, B, 0\}$$

einteilen. Die Anteile der Blutgruppen AB, A, B und 0 (gegeben durch f_B) werden mit 5%, 43%, 11% beziehungsweise 41% angenommen und entsprechen damit der Verteilung Deutschlands im Jahr 2020 [12]. Die angesetzte Verteilung der Altersklassen (f_A) ist ebenfalls ein Abbild der Situation in Deutschland, bezieht sich jedoch auf das Jahr 2022 [20]. In Abbildung 5.1 finden sich Darstellungen beider Verteilungen.



(a) Blutgruppenverteilung f_B

(b) Altersklassenverteilung f_A

Abbildung 5.1: Angenommene Bevölkerungsstruktur hinsichtlich Altersklassen- und Blutgruppenverteilung.

5.1.1.2 Fixierte Anfangswerte

In allen vier Szenarien beginnt das Infektionsgeschehen mit genau einem infizierten Individuum aus der Altersklasse $25 - 29$ Jahre mit Blutgruppe A. Alle restlichen Individuen werden als suszeptibel angenommen. Die feste Wahl der Altersklasse und Blutgruppe des initialen Infektionsfalls ist zwar willkürlich getroffen, dient aber der Vergleichbarkeit ver-

schiedener Konfigurationen.

$$I(0, a_i, b_j) = \begin{cases} 1 & a_i = 25 - 29 Jahre,\ b_j = A \\ 0 & \text{sonst} \end{cases}$$

$$S(0, a_i, b_j) = \begin{cases} N \cdot f_A(a_i) \cdot f_B(b_j) - 1 & a_i = 25 - 29 Jahre,\ b_j = A \\ N \cdot f_A(a_i i) \cdot f_B(b_j) & \text{sonst} \end{cases}$$

5.1.1.3 Fixierte Genesungsrate

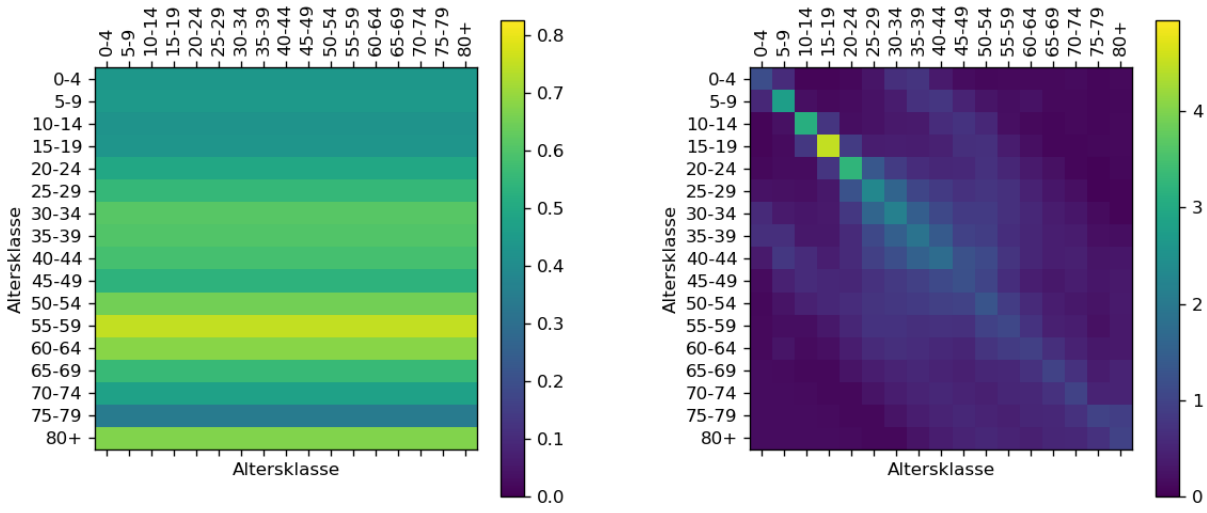
Die tägliche Genesungsrate $\gamma \in \mathbb{R}^{17}$ wird für alle Szenarien dieses Abschnitts als homogen mit

$$\gamma[i] = \frac{1}{6} \quad \text{für } i = 1, 2, \ldots, 17$$

angenommen, was einer durchschnittlichen Krankheitsdauer von sechs Tagen entspricht.

5.1.1.4 Variierte Kontaktmatrix

Für die Kontaktmatrix κ werden zwei Belegungen κ_{Zufall} und κ_{Muster}, die in Abbildung 5.2 dargestellt werden, unterschieden. Die Kontaktmatrix κ_{Muster} basiert auf einer Studie aus dem Vereinigten Königreich, die altersspezifische Kontaktmuster mittels einer per App durchgeführten Personenbefragung untersucht hat [10].



(a) Heatmap der Kontaktmatrix κ_{Zufall}

(b) Heatmap Kontaktmatrix κ_{Muster}

Abbildung 5.2: Abbildung der zwei Varianten der Kontaktmatrix κ.

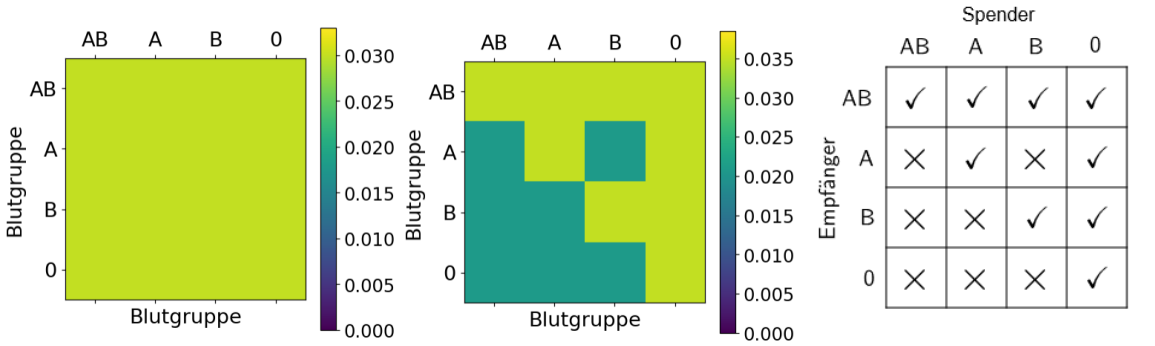
Gemäß κ_{Muster} variieren die täglichen Kontakte der Individuen unterschiedlicher Altersklassen, wobei die durchschnittliche tägliche Kontaktanzahl bei $\bar{\kappa}_{\text{Muster}} = 9.48$ liegt. In κ_{Zufall} hat hingegen jedes Individuum 9.48 tägliche Kontakte ($\Rightarrow \bar{\kappa}_{\text{Zufall}} = 9.48$), die sich auf die 17 Altersklassen entsprechend deren Anteile aufteilen:

$$\kappa_{\text{Zufall}}[i, j] = 9.48 \cdot f_A(a_i)$$

Somit lassen sich die in κ_{Zufall} repräsentierten Kontakte als rein zufällig oder auf einer uniformen Durchmischung der Population basierend interpretieren. Die beiden Varianten κ_{Muster} und κ_{Zufall} unterscheiden sich zusammenfassend also nur in der Verteilung der täglichen Kontakte, nicht aber in der Gesamtzahl an täglichen Kontakten innerhalb der Population.

5.1.1.5 Variierte Wahrscheinlichkeitsmatrix

Die Wahrscheinlichkeitsmatrix wird in den beiden Ausprägungen β_{homo} und β_{inhomo} betrachtet (siehe Abbildung 5.3). Die Variante β_{homo} repräsentiert eine homogene Ansteckungswahrscheinlichkeit von 3% bei Kontakt zwischen einem suszeptiblen und einem infizierten Individuum ($\Rightarrow \bar{\beta}_{\text{homo}} = 3\%$). Die zweite Variante β_{inhomo} modelliert die gegenläufige Annahme, dass die Ansteckungswahrscheinlichkeit davon abhängt, ob die Blutgruppe des infizierten Individuums mit der des suszeptiblen Individuums nach dem AB0-Blutgruppensystem kompatibel ist: In Fällen von Blutgruppenkompatibilität wird eine Ansteckungswahrscheinlichkeit von $3,5\%$ angesetzt, während bei Inkompatibilität der Blutgruppen eine Wahrscheinlichkeit von $2,1\%$ angenommen wird, was einer Reduktion um 40% entspricht. Unter oben definierter Blutgruppenverteilung ergibt dies eine durchschnittliche Ansteckungswahrscheinlichkeit von $\bar{\beta}_{\text{inhomo}} = 2,99\% \approx 3\%$. Demzufolge unterscheiden sich β_{homo} und β_{inhomo} ausschließlich in der Verteilung der Ansteckungswahrscheinlichkeit.



(a) Heatmap der Wahrscheinlichkeitsmatrix β_{homo}

(b) Heatmap der Wahrscheinlichkeitsmatrix β_{inhomo}

(c) Kompatibilität nach dem AB0-Blutgruppensystem

Abbildung 5.3: Gegenüberstellung der zwei Varianten der Wahrscheinlichkeitsmatrix β.

5.1.2 Konfiguration der Lösungsverfahren

Als letzter Teil der Experimentkonfiguration ist für beide Lösungsverfahren eine Zeitschrittweite Δt festzulegen. Exklusiv für das rangadaptive Eulerverfahren gilt es zusätzlich, die den erlaubten Kürzungsfehler mitbestimmenden Konstanten M_1 und M_2 zu definieren (vgl. Theorem 4.2). Die gewählten Optionen finden sich in Tabelle 5.1 zusammengefasst.

Lösungsverfahren	Freie Parameter	Abhängige Parameter
Explizites Eulerverfahren	$\Delta t = \frac{1}{2}$	-
Rangadaptives Eulerverfahren	$\Delta t = \frac{1}{2}$	
	$M_1 = 1 \times 10^{-1}$	$\epsilon_r = \frac{M_1 \Delta t}{4} = 1.25 \times 10^{-4}$
	$M_2 = 1 \times 10^{-1}$	$\epsilon_k = \frac{M_2 \Delta t^2}{2} = 1.25 \times 10^{-4}$

Tabelle 5.1: Freie und abhängige Parameter des expliziten Eulerverfahrens und des rangadaptiven Eulerverfahrens für die Berechnungen in Kapitel 5.1.

5.1.3 Ergebnisse

In diesem Abschnitt wird erläutert, wie sich die vier möglichen Parameterbelegungen der Kontakt- und Wahrscheinlichkeitsmatrix gegeben durch

$$
\begin{aligned}
&1)\ \kappa_{\text{Zufall}} \times \beta_{\text{homogen}} \\
&2)\ \kappa_{\text{Zufall}} \times \beta_{\text{inhomogen}} \\
&3)\ \kappa_{\text{Muster}} \times \beta_{\text{homogen}} \\
&4)\ \kappa_{\text{Muster}} \times \beta_{\text{inhomogen}}
\end{aligned}
$$

unter Konstanthaltung der restlichen Experimentkonfiguration auf einerseits die Lösung des erweiterten SIR-Modells und andererseits die Eigenschaften des rangadaptiven Eulerverfahrens im Vergleich zum expliziten Eulerverfahren auswirken. Zunächst soll jedoch aufgegriffen werden, dass für die Mittelwerte der jeweils zwei Parameterausprägungen $\bar{\kappa} = \bar{\kappa}_{\text{Zufall}} = \bar{\kappa}_{\text{Muster}} = 9.48$ und $\bar{\beta} = \bar{\beta}_{\text{homogen}} = \bar{\beta}_{\text{inhomogen}} = 3\%$ gilt. Daher ist zu betonen, dass die festgestellten Unterschiede zwischen den vier Parameterkombinationen ausschließlich auf unterschiedliche Verteilungen der Kontakte und der Ansteckungswahrscheinlichkeit zurückzuführen sind. Es handelt sich nicht um insgesamt verschiedene Anzahlen von Kontakten oder verschiedene mittlere Ansteckungswahrscheinlichkeiten.

Vergleich zum klassischem SIR-Modell: Als erstes wird untersucht, wie sich die vier verschiedenen Konfigurationen auf den makroskopischen Verlauf der Lösung des erweiterten SIR-Modells auswirken (siehe Abbildung 5.4). Als Bezugspunkt dient in dieser Sache die Lösung eines klassischen SIR-Modells, dessen Parameter $\bar{\lambda} = \bar{\kappa} \cdot \bar{\beta} = 9.48 \cdot 0.03 = 0.2844$

und $\bar{\gamma} = 1/6$ auf den Mittelwerten der Parameter des erweiterten SIR-Modells beruhen. Die zugehörigen Anfangswerte $\tilde{S}^0 = 9999$ und $\tilde{I}^0 = 1$ entsprechen den über alle Altersklassen und Blutgruppen summierten Anfangswerten des erweiterten SIR-Modells. Es zeigt sich, dass die Lösung des erweiterten SIR-Modells unter der Kombination $\kappa_{\text{Zufall}} \times \beta_{\text{homo}}$ mit der Lösung des klassischen SIR-Modells übereinstimmt. Dieses Verhalten lässt sich damit erklären, dass das erweiterte SIR-Modell mit κ_{Zufall} und β_{homo} mathematisch nicht vom klassischen SIR-Modell zu unterscheiden ist. Ferner ist bemerkenswert, dass die Lösungen der anderen Konfigurationen im Vergleich zur Lösung des klassischen SIR-Modells nur als leicht auf der x-Achse verschoben erscheinen.

Blutgruppenstratifizierte Prävalenz: In Abbildung 5.5 ist für jede der vier unterschiedlichen Parameterkonfigurationen die relative Prävalenz der verschiedenen Blutgruppen im zeitlichen Verlauf dargestellt. Es ist zu sehen, dass die Einführung der blutgruppenabhängigen Ansteckungswahrscheinlichkeit β_{inhomo} dazu führt, dass die Verläufe der relativen Prävalenzen der unterschiedlichen Blutgruppen unterschiedlich ausfallen. Ein Vergleich der maximalen relativen Prävalenz mit der durchschnittlichen Ansteckungswahrscheinlichkeit einer Blutgruppe zeigt, dass beide Größen positiv korrelieren (siehe Tabelle 5.2). Dieser Tabelle ist des Weiteren zu entnehmen, dass die Kontaktmatrix κ_{Muster} in allen Fällen mit einer höheren (relativen) Maximalprävalenz einhergeht.

Blutgruppe	Mittlere Ansteckungswahrscheinlichkeit	Maximale relative Prävalenz	Konfiguration
AB	3.5%	10.8%	$\kappa_{\text{Zufall}} \times \beta_{\text{inhomo}}$
		11.9%	$\kappa_{\text{Muster}} \times \beta_{\text{inhomo}}$
A	3.2%	10.4%	$\kappa_{\text{Zufall}} \times \beta_{\text{inhomo}}$
		11.5%	$\kappa_{\text{Muster}} \times \beta_{\text{inhomo}}$
B	2.8%	9.5%	$\kappa_{\text{Zufall}} \times \beta_{\text{inhomo}}$
		10.5%	$\kappa_{\text{Muster}} \times \beta_{\text{inhomo}}$
0	2.7%	9.1%	$\kappa_{\text{Zufall}} \times \beta_{\text{inhomo}}$
		10.1%	$\kappa_{\text{Muster}} \times \beta_{\text{inhomo}}$

Tabelle 5.2: Gegenüberstellung der maximalen relativen Prävalenz und durchschnittlichen Ansteckungswahrscheinlichkeit pro Blutgruppe unter der Wahl von β_{inhomo} als Wahrscheinlichkeitsmatrix.

Altersklassenstratifizierte Prävalenz: Analog zu den vorangegangenen Erläuterungen beinhaltet Abbildung 5.6 die relativen Prävalenzen fünf exemplarisch ausgewählter Altersklassen im zeitlichen Verlauf. Diesmal bedingt die Einführung der altersabhängigen Kon-

taktmatrix κ_{Muster} ein Auffächern der relativen Prävalenzen der unterschiedlichen Altersklassen. Eine Gegenüberstellung der täglichen Kontakte und maximaler relativer Prävalenz pro Altersklasse findet sich in Tabelle 5.3. An ihr ist abzulesen, dass Individuen aus der Altersklasse $0-4$ Jahre zwar weniger tägliche Kontakte als Individuen aus der Altersklasse 80+ Jahre haben, die (maximale) relative Prävalenz unter den zuerst genannten aber trotzdem höher ausfällt. Eine mögliche Erklärung für dieses zunächst kontraintuitive Ergebnis bietet der Fakt, dass Individuen aus der Altersklasse $0-4$ Jahre zwar insgesamt weniger tägliche Kontakte haben, jedoch mehr tägliche Kontakte zu Personen mittleren Alters (vgl. Abbildung 5.2). Die Personen mittleren Alters haben wiederum selbst besonders viele tägliche Kontakte, sodass das mittelbare Kontaktnetzwerk für Individuen aus der Altersklasse $0-4$ Jahre größer ausfällt als für Individuen aus der Altersklasse 80+ Jahre.

Altersklasse	Tägliche Kontakte	Maximale relative Prävalenz	Konfiguration
$0-4$ Jahre	4.9	8.6%	$\kappa_{\text{Muster}} \times \beta_{\text{homo}}$
		8.2%	$\kappa_{\text{Muster}} \times \beta_{\text{inhomo}}$
$20-24$ Jahre	10.5	13.4%	$\kappa_{\text{Muster}} \times \beta_{\text{homo}}$
		12.9%	$\kappa_{\text{Muster}} \times \beta_{\text{inhomo}}$
$40-44$ Jahre	13	13.8%	$\kappa_{\text{Muster}} \times \beta_{\text{homo}}$
		13.3%	$\kappa_{\text{Muster}} \times \beta_{\text{inhomo}}$
$60-64$ Jahre	8	9.5%	$\kappa_{\text{Muster}} \times \beta_{\text{homo}}$
		9.1%	$\kappa_{\text{Muster}} \times \beta_{\text{inhomo}}$
80+ Jahre	5.5	7.2%	$\kappa_{\text{Muster}} \times \beta_{\text{homo}}$
		7%	$\kappa_{\text{Muster}} \times \beta_{\text{inhomo}}$

Tabelle 5.3: Gegenüberstellung der maximalen relativen Prävalenz und täglichen Kontaktanzahl pro Altersklasse unter der Wahl von κ_{Muster}.

Fehler: Wird die mit dem expliziten Eulerverfahren (EE) berechnete Lösung (S_{EE}, I_{EE}) als Referenzlösung herangezogen, findet sich der absolute Fehler der mit dem rangadaptiven Eulerverfahren (RE) berechneten Lösung (S_{RE}, I_{RE}) in Abbildung 5.7 dargestellt. Dieser ist unter jeder Konfiguration und zu jedem Zeitpunkt kleiner als 0.5, wobei er für S_{RE} stets größer ausfällt als für I_{RE}. In jedem Fall strebt der Fehler von I_{RE} gegen Ende des berechneten Zeitraums wieder gegen null, während der Fehler von S_{RE} konfigurationsabhängig gegen einen Wert im Bereich zwischen 0.12 und 0.16 strebt.

Speicherbedarf: Als nächster Aspekt wird untersucht, wie viel Speicherplatz die hierarchischen Tuckertensoren der mit dem rangadaptiven Eulerverfahren berechneten Lösungen

im Vergleich zu den vollen Tensoren der mit dem expliziten Eulerverfahren berechneten Lösungen einnehmen. Der von der gesamten Lösung benötigte Speicherbedarf, gegeben durch $\sum_n \text{Speicher}\,(S^n, I^n)$, ist für die unterschiedlichen Konfigurationen in Tabelle 5.4 zu sehen. Der Speicherbedarf der Lösungen der verschiedenen Zeitpunkte, jeweils gegeben durch $\text{Speicher}\,(S^n, I^n)$, ist hingegen in Abbildung 5.8 dargestellt. Der summierte Speicherbedarf der Lösungen mit hierarchischen Tuckertensoren fällt in allen betrachteten Konfigurationen verglichen mit den vollen Tensoren um etwa $30 - 50\%$ geringer aus. Während ein Variieren der Wahrscheinlichkeitsmatrix kaum Einfluss auf den Bedarf an Speicherplatz ausübt, führt ein Übergang von κ_{Zufall} auf κ_{Muster} zu einem Ansteigen des eingenommenen Speicherbedarfs um grob $15 - 20\%$.

Gesamtspeicherbedarf in KB		κ_{Zufall}	κ_{Muster}
β_{homo}	$\sum_n \text{Speicher}\,(S^n_{RE}, I^n_{RE})$	119.71	143.33
	$\sum_n \text{Speicher}\,(S^n_{EE}, I^n_{EE})$	218.69	218.69
β_{inhomo}	$\sum_n \text{Speicher}\,(S^n_{RE}, I^n_{RE})$	120.29	146.21
	$\sum_n \text{Speicher}\,(S^n_{EE}, I^n_{EE})$	218.69	218.69

Tabelle 5.4: Darstellung des Gesamtspeicherbedarfs der mit dem rangadaptiven Eulerverfahren (RE) und expliziten Eulerverfahren (EE) berechneten Lösungen des erweiterten SIR-Modells unter der in Kapitel 5.1 angelegten Konfiguration bei Variation der Kontaktmatrix κ und der Wahrscheinlichkeitsmatrix β.

Hierarchischer Rang: Der maximale hierarchische Rang der mit dem rangadaptiven Eulerverfahren berechneten Lösung (S_{RE}, I_{RE}) unterscheidet sich in allen vier betrachteten Konfigurationen für die Komponenten S_{RE} und I_{RE} (siehe Abbildung 5.9). Für S_{RE} liegt er in allen Fällen konstant bei zwei, während er für I_{RE} zwischen eins und zwei pendelt und dabei das gleiche Muster wie der Speicherbedarf zeigt (vgl. Abbildung 5.8). Dass die Entwicklung des maximalen hierarchischen Ranges und die des Speicherbedarfs korrelieren, entspricht der Erwartung, da der Speicherbedarf direkt vom hierarchischen Rang abhängt (siehe Lemma 2.39).

Berechnungsgeschwindigkeit: Unter der gegebenen Experimentkonfiguration ist die Berechnungsgeschwindigkeit ein zentraler Nachteil des rangadaptiven Eulerverfahrens im Vergleich zum expliziten Eulerverfahren. Diese unterscheidet sich zu Ungunsten des rangadaptiven Eulerverfahrens um zwei Größenordnungen (siehe Tabelle 5.5). Während die Wahl der Wahrscheinlichkeitsmatrix die Berechnungsgeschwindigkeit nicht beeinflusst, ist eine Absenkung der Berechnungsgeschwindigkeit um $\sim 10\%$ bei der Wahl von κ_{Muster} gegenüber κ_{Zufall} zu verzeichnen. Als Ursache für diese Absenkung ist der erhöhte hierarchische Rang

unter κ_{Muster} auszumachen (vgl. Abbildung 5.9).

Berechnungsgeschwindigkeit in it/s		κ_{Zufall}	κ_{Muster}
β_{homo}	Rangadaptives Eulerverfahren	156.8 ± 0.74	140.9 ± 1.1
	Explizites Eulerverfahren	11057 ± 321	11043 ± 305
β_{inhomo}	Rangadaptives Eulerverfahren	156.7 ± 0.56	139.5 ± 0.62
	Explizites Eulerverfahren	11038 ± 310	11053 ± 308

Tabelle 5.5: Gegenüberstellung der durchschnittlichen Berechnungsgeschwindigkeiten des rangadaptiven Eulerverfahrens und des expliziten Eulerverfahrens unter der in Kapitel 5.1 festgelegten Experimentkonfiguration bei Variation der Kontaktmatrix κ und der Wahrscheinlichkeitsmatrix β.

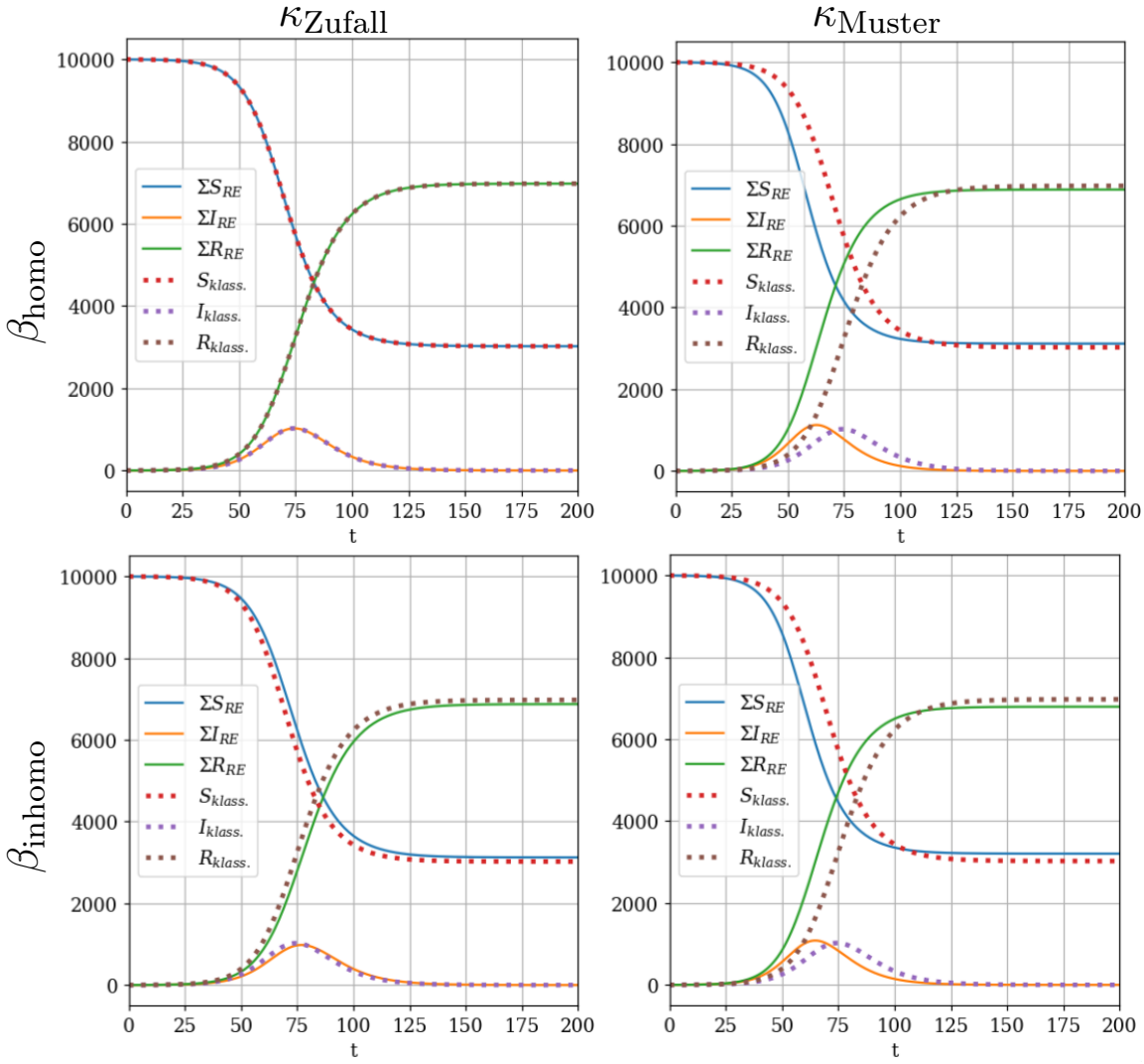


Abbildung 5.4: Unter Konstanthaltung der in 5.1 beschriebenen Experimentkonfiguration wird für jede der vier Kombinationen aus κ_{Zufall}, κ_{Muster} und β_{homo}, β_{inhomo} die mithilfe des rangadaptiven Eulerverfahrens (RE) berechnete Lösung des resultierenden erweiterten SIR-Modells abgebildet. Die dargestellten Kurven ergeben sich hierbei durch Summieren der suszeptiblen, infizierten und immunen Individuen aller Altersklassen und Blutgruppen. Als Referenzlösung dient die Lösung eines klassischen SIR-Modells, dessen Parameter sich aus den übereinstimmenden Parametermittelwerten der betrachteten Parameterkonfigurationen des erweiterten SIR-Modells ergeben.

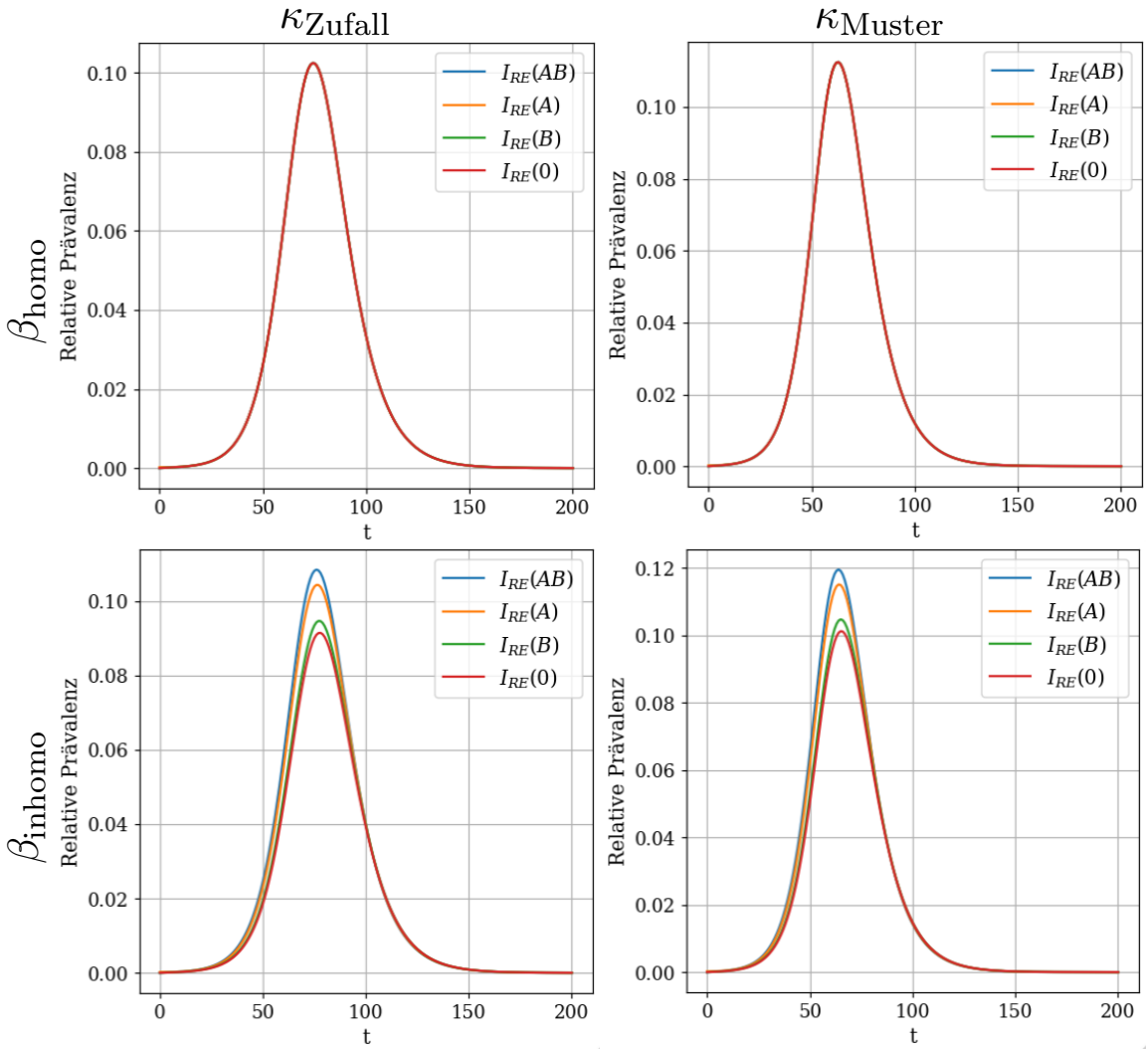


Abbildung 5.5: Darstellung der zeitlichen Verläufe der relativen Prävalenzen pro Blutgruppe für jede der vier Kombinationen aus κ_{Zufall}, κ_{Muster} und β_{homo}, β_{inhomo}. Die relativen Prävalenzen basieren auf den Lösungen des erweiterten SIR-Modells, die unter der in 5.1 beschriebenen Experimentkonfiguration mithilfe des rangadaptiven Eulerverfahrens (RE) berechnet wurden.

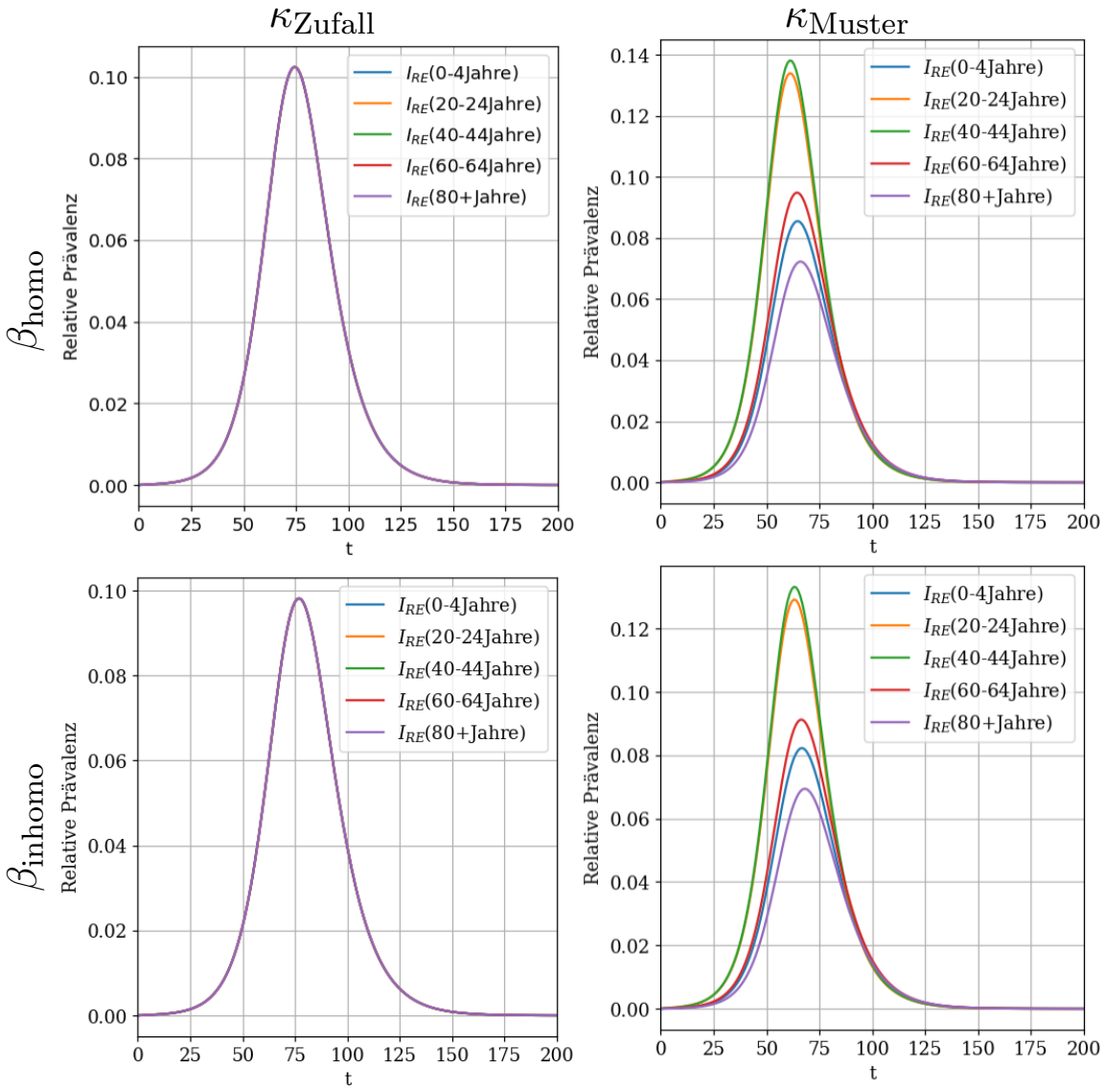


Abbildung 5.6: Darstellung der zeitlichen Verläufe der relativen Prävalenzen für fünf ausgewählte Altersklassen. Jedes der vier abgebildeten Diagramme spiegelt eine der vier Kombinationen aus κ_{Zufall}, κ_{Muster} und β_{homo}, β_{inhomo} wider. Die relativen Prävalenzen basieren auf den Lösungen des erweiterten SIR-Modells, die unter der in 5.1 beschriebenen Experimentkonfiguration mithilfe des rangadaptiven Eulerverfahrens (RE) berechnet wurden.

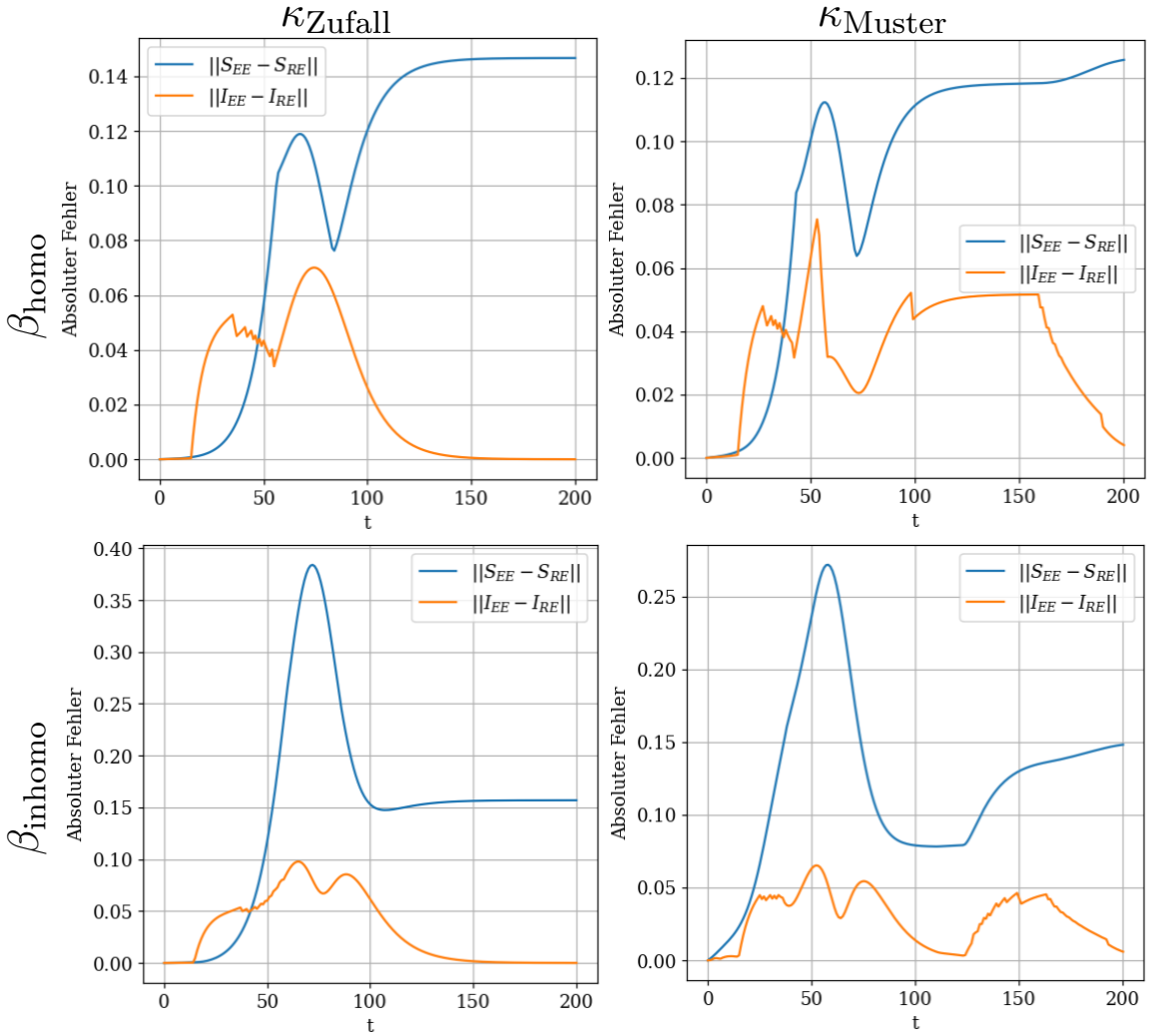


Abbildung 5.7: In dieser Abbildung ist der absolute Fehler der auf dem rangadaptiven Eulerverfahren (RE) basierenden Lösung dargestellt. Als Referenzlösung dient die mit dem expliziten Eulerverfahren (EE) berechnete Lösung. Sämtliche Lösungen beziehen sich auf das erweiterte SIR-Modell unter der in 5.1 definierten Konfiguration bei Variation der Kontaktmatrix κ und der Wahrscheinlichkeitsmatrix β.

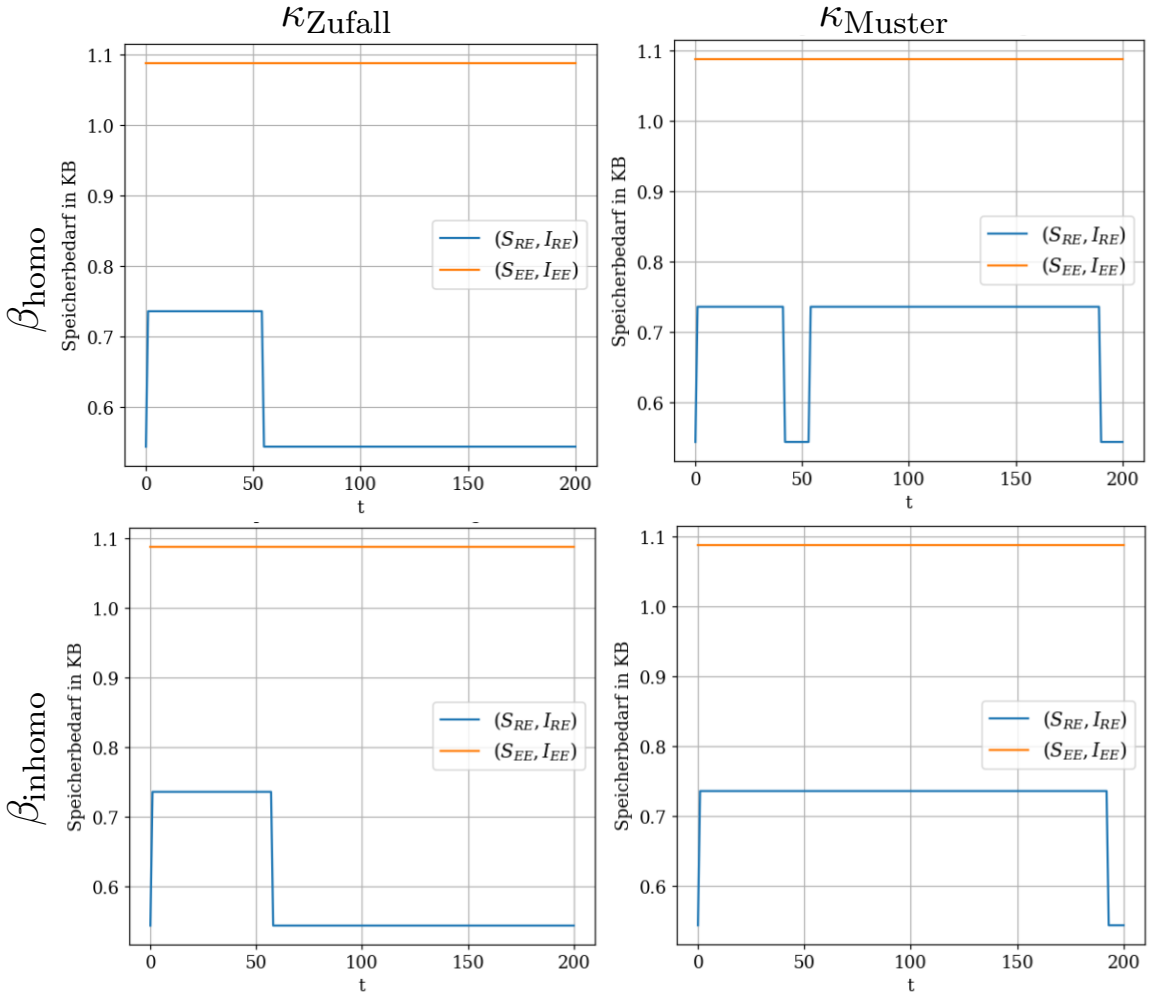


Abbildung 5.8: Die vier Diagramme dieser Abbildung vergleichen den Speicherbedarf der mit dem rangadaptiven Eulerverfahren (RE) und expliziten Eulerverfahren (EE) berechneten Lösungen des erweiterten SIR-Modells unter der in 5.1 festgelegten Konfiguration bei Variation der Kontaktmatrix κ und der Wahrscheinlichkeitsmatrix β.

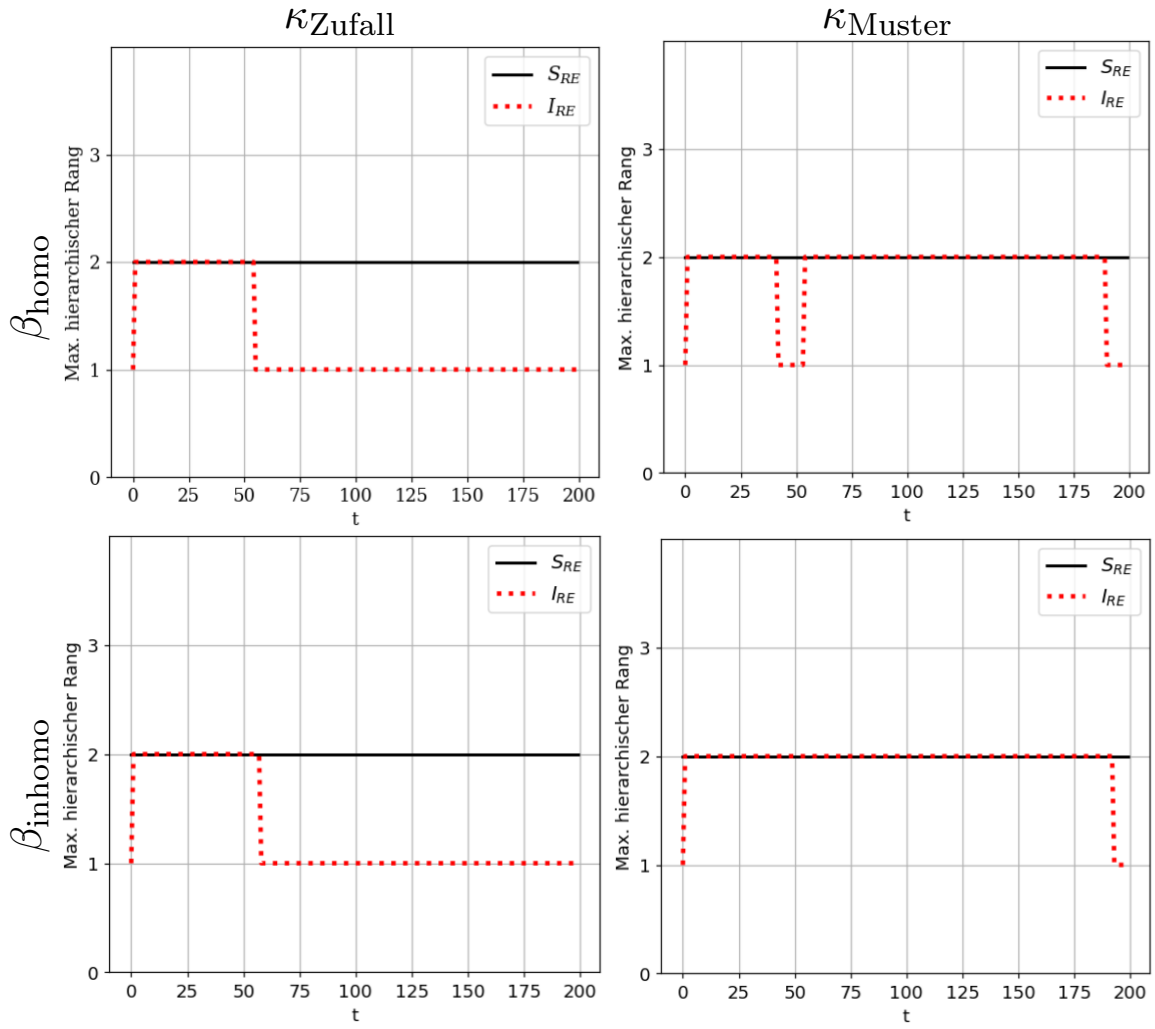


Abbildung 5.9: Die Diagramme dieser Abbildung präsentieren den zeitlichen Verlauf der maximalen hierarchischen Ränge der Lösungen des erweiterten SIR-Modells, berechnet mit dem rangadaptiven Eulerverfahren (RE). Die Berechnungen erfolgten unter Verwendung der in Abschnitt 5.1 definierten Konfiguration, wobei sowohl die Kontaktmatrix κ als auch die Wahrscheinlichkeitsmatrix κ variiert wurden.

5.2 Reaktion und Diffusion

In diesem Kapitel erfolgen Berechnungen auf Grundlage des vollständigen erweiterten SIR-Modells. Im Vergleich zum vorherigen Kapitel entspricht dies der Hinzunahme der räumlichen Auflösung beziehungsweise dem Übergang hin zu partiellen Differentialgleichungen. Es werden vier exemplarische Szenarien mit einer fixierten Basiskonfiguration unter Variation der Gitterweite $\Delta x = \Delta y = h$ und des Diffusionskoeffizienten D berechnet. Der erste Teil dieses Kapitels ist der detaillierten Erläuterung der vier Experimentkonfigurationen gewidmet, während im zweiten Teil die Ergebnisse präsentiert werden.

5.2.1 Modellkonfiguration

5.2.1.1 Fixierte Bevölkerungsstruktur

In allen vier Experimenten wird eine Bevölkerung mit $N = 10^7$ Individuen angenommen, die jeweils einer der 17 Altersklassen

$$\mathcal{A} = \{0 - 4Jahre, 5 - 9Jahre, \ldots 75 - 79Jahre, 80 + Jahre\}$$

und einer der vier Blutgruppen

$$\mathcal{B} = \{AB, A, B, 0\}.$$

angehören. Wie in der vorherigen Experimentkonfiguration orientieren sich die angenommenen Verteilungen der Altersklassen und Blutgruppen (f_A beziehungsweise f_B) an denen in Deutschland (siehe Abbildung 5.1). Die von der Population besiedelte Fläche wird dabei exemplarisch als Einheitsquadrat

$$\Omega = [0, 1] \times [0, 1]$$

angenommen. Der in der Formulierung des Modells aufgestellten Voraussetzung folgend, ist die Bevölkerungsdichte homogen mit $f_\Omega(x, y) = \frac{N}{\mu(\Omega)} = N$.

5.2.1.2 Fixierte Anfangswerte

Alle vier Szenarien starten mit genau einem infizierten Individuum aus der Altersklasse $25-29$ Jahre mit Blutgruppe A, das sich im Mittelpunkt der betrachteten Fläche Ω befindet. Die Anfangswerte sind demnach wie folgt gegeben:

$$I(0, a_i, b_j, x, y) = \begin{cases} 1 & a_i = 25 - 29Jahre,\ b_j = A,\ x = y = 0.5 \\ 0 & \text{sonst} \end{cases}$$

$$S(0, a_i, b_j, x, y) = \begin{cases} N f_A(a_i) f_B(b_j) - 1 & a_i = 25 - 29 Jahre,\ b_j = A,\ x = y = 0.5 \\ N f_A(a_i) f_B(b_j) & \text{sonst} \end{cases}$$

5.2.1.3 Fixierte Modellparameter

Die drei Modellparameter γ, κ und β werden für alle vier Simulationen mit den in Tabelle 5.6 dargelegten Belegungen versehen.

Modellparameter	Belegung
Tägl. Genesungsrate	$\gamma[i] = \frac{1}{6}$ für $i = 1, \ldots, 17$
Kontaktmatrix	$\kappa = \kappa_{\text{Muster}}$
Wahrscheinlichkeitsmatrix	$\beta = \beta_{\text{inhomo}}$
Diffusionskoeffizient	$D \in \{4 \times 10^{-5},\ 2 \times 10^{-5}\}$

Tabelle 5.6: Belegungen der Modellparameter γ, κ, β und D für die Simulationen in Kapitel 5.2. Die Definitionen für κ_{Muster} und β_{inhomo} finden sich in Unterabschnitt 5.1.1.4 und 5.1.1.5.

5.2.1.4 Variierter Diffusionskoeffizient

Der Diffusionskoeffizient ist der einzige Modellparameter, der für die vier Experimente variiert wird. Er nimmt hierbei die Werte $D = 4 \times 10^{-5}$ und $D = 2 \times 10^{-5}$ an.

5.2.2 Konfiguration der Lösungsverfahren

Für beide Lösungsverfahren gilt es, die Zeitschrittweite Δt und die Gitterweite $\Delta x = \Delta y = h$ festzulegen. Für das rangadaptive Eulerverfahren müssen zusätzlich Belegungen für die Konstanten M_1 und M_2, die die Fehlertoleranzen ϵ_r beziehungsweise ϵ_k mitbestimmen, definiert werden. Die gewählten Optionen finden sich in Tabelle 5.7 zusammengefasst.

Lösungsverfahren	Freie Parameter	Abhängige Parameter
Explizites Eulerverfahren	$\Delta t = 1/32$ $h \in \{1/128,\ 1/256\}$	-
Rangadaptives Eulerverfahren	$\Delta t = 1/32$ $M_1 = 1 \times 10^{-1}$ $M_2 = 1 \times 10^{-1}$ $h \in \{1/128,\ 1/256\}$	$\epsilon_r = \frac{M_1 \Delta t}{4} \approx 7.8 \times 10^{-4}$ $\epsilon_k = \frac{M_2 \Delta t^2}{2} \approx 4.9 \times 10^{-5}$

Tabelle 5.7: Freie und abhängige Parameter des expliziten Eulerverfahrens und des rangadaptiven Eulerverfahrens für die Berechnungen in Kapitel 5.2.

5.2.3 Ergebnisse

Dieser Abschnitt ist der Diskussion gewidmet, wie sich die vier möglichen Kombinationen der unterschiedlichen Belegungen des Diffusionskoeffizienten D und der Gitterweite h gegeben durch

$$\begin{aligned} &1)\ D = 2 \times 10{-}5 \text{ und } h = \frac{1}{256} \\ &2)\ D = 2 \times 10{-}5 \text{ und } h = \frac{1}{128} \\ &3)\ D = 4 \times 10{-}5 \text{ und } h = \frac{1}{256} \\ &4)\ D = 4 \times 10{-}5 \text{ und } h = \frac{1}{128} \end{aligned}$$

unter Konstanthaltung der übrigen Experimentkonfiguration auf einerseits die Lösung des erweiterten SIR-Modells und andererseits das rangadaptive Eulerverfahren gemessen am expliziten Eulerverfahren auswirken.

Makroskopischer Verlauf: Die Entwicklung der Gesamtzahlen an suszeptiblen, infizierten und immunen Individuen zeigt in allen vier Szenarien einen epidemischen Verlauf (siehe Abbildung 5.10), der stets im selben Gleichgewichtszustand zu enden scheint. Während die Verdopplung des Diffusionskoeffizienten von 2×10^{-5} auf 4×10^{-5} den Gesamtverlauf durch eine leichte Zuspitzung der Prävalenzkurve beeinflusst, wirkt sich die unterschiedliche Gitterweite nicht auf den Gesamtverlauf aus. Ferner kann festgestellt werden, dass unter den gegebenen Konfigurationen die Lösungen des rangadaptiven Eulerverfahrens und des expliziten Eulerverfahrens nahezu identisch verlaufen.

Räumlicher Verlauf: In allen vier Szenarien breitet sich die Krankheit beginnend beim ersten Infizierten wellenartig in den Raum aus. Der Lösungsverlauf entspricht dabei immer größer werdenden konzentrischen Ringen, die den Ort des ersten Infizierten als Mittelpunkt haben (vgl. Tabelle 5.10). Die Existenz derartiger Wanderwellen in SIR-Modellen mit räumlicher Auflösung wird beispielsweise in [8] und [23] näher untersucht.

Fehler: Der zeitliche Verlauf des absoluten Fehlers der mittels rangadaptiven Eulerverfahren berechneten Lösungen (S_{RE}, I_{RE}) gemessen an den durch das explizite Eulerverfahren ermittelten Lösungen (S_{EE}, I_{EE}) findet sich in Abbildung 5.11. Wie im Experiment des vorherigen Kapitels dominiert für jede der vier Konfigurationen der Fehler von S_{RE}. Des Weiteren ist zu beobachten, dass der Fehler für S_{RE} und I_{RE} parallel zum Ausbreiten der oben erwähnten Wanderwelle ansteigt und mit dem Übergang in den stationären Zustand wieder abfällt. Davon abgesehen stellt sich heraus, dass die kleinere beider Gitterweiten mit einem geringeren Fehler der Lösung assoziiert ist, während für die beiden Belegungen des Diffusionskoeffizienten kein eindeutiger Einfluss ausmachbar ist.

Speicherbedarf: Als nächstes wird erläutert, wie viel Speicherplatz die hierarchischen Tuckertensoren der mit dem rangadaptiven Eulerverfahren berechneten Lösungen im Ver-

gleich zu den vollen Tensoren der mit dem expliziten Eulerverfahren berechneten Lösungen einnehmen. Für alle unterschiedlichen Konfigurationen ist der jeweils gesamte Speicherbedarf der Lösung, gegeben durch $\sum_n \text{Speicher}\,(S^n, I^n)$, in Tabelle 5.8 zu finden. Der Speicherbedarf der Lösungen der verschiedenen Zeitpunkte, jeweils gegeben durch Speicher (S^n, I^n), ist hingegen in Abbildung 5.12 dargestellt. Die bedeutendste Beobachtung und gleichzeitig ein Beleg für die Leistungsfähigkeit des hierarchischen Tuckerformats besteht darin, dass die Speicheranforderungen der Lösungen im hierarchischen Tuckerformat in allen vier Konfigurationen um zwei bis drei Größenordnungen geringer ausfallen als die der Lösungen mit vollen Tensoren. Die Stärke der hierarchischen Tuckertensoren zeigt sich des Weiteren bei einer Halbierung der Gitterweite. Diese vervierfacht den benötigten Speicherplatz der Lösungen mit vollen Tensoren, während sie den Speicherbedarf der Lösungen mit hierarchischen Tuckertensoren nur um jeweils $\sim 50\%$ anhebt.

Gesamtspeicherbedarf		$h = \frac{1}{128}$	$h = \frac{1}{256}$
$D = 2 \times 10^{-5}$	$\sum_n \text{Speicher}\,(S^n_{RE}, I^n_{RE})$	$21.99MB$	$32.9MB$
	$\sum_n \text{Speicher}\,(S^n_{EE}, I^n_{EE})$	$5.45GB$	$21.63GB$
$D = 4 \times 10^{-5}$	$\sum_n \text{Speicher}\,(S^n_{RE}, I^n_{RE})$	$18.24MB$	$27.5MB$
	$\sum_n \text{Speicher}\,(S^n_{EE}, I^n_{EE})$	$5.45GB$	$21.63GB$

Tabelle 5.8: Darstellung des Gesamtspeicherbedarfs der mit dem rangadaptiven Eulerverfahren (RE) und expliziten Eulerverfahren (EE) berechneten Lösungen des erweiterten SIR-Modells unter der in Kapitel 5.2 angelegten Konfiguration bei Variation des Diffusionskoeffizienten D und der Gitterweite h.

Hierarchischer Rang: Die zeitliche Entwicklung der maximalen hierarchischen Ränge der hierarchischen Tuckertensoren der mit dem rangadaptiven Eulerverfahren berechneten Lösungen ist für die vier betrachteten Konfigurationen dieses Kapitel in Abbildung 5.13 ersichtlich. In allen vier Szenarien zeigt der maximale hierarchische Rang von S_{RE} und I_{RE} ein stufenweises Anwachsen bis zu einem jeweils konfigurationsabhängigen Maximum von etwa 15 ± 3, das anschließend gehalten wird. Das jeweilige Maximum wird in den beiden Konfigurationen mit $D = 2 \times 10^{-5}$ erst bei $t \approx 200$ erreicht, während es in den Konfigurationen mit $D = 4 \times 10^{-5}$ bereits bei $t \approx 150$ erreicht wird. Abgesehen davon zeigt sich, wie zu erwarten, eine deutliche Korrelation zwischen dem maximalen hierarchischen Rang einer Lösung zu einem bestimmten Zeitpunkt und dem zu diesem Zeitpunkt erforderlichen Speicherplatzbedarf für diese Lösung.

Berechnungsgeschwindigkeit: Die durchschnittlichen Berechnungsgeschwindigkeiten beider Verfahren werden in Tabelle 5.9 präsentiert. Anders als in den Simulationen des vorangegangenen Kapitels fällt die Berechnungsgeschwindigkeit diesmal zugunsten des ranga-

daptiven Eulerverfahrens aus. Ferner ist der Tabelle zu entnehmen, dass unter Konstanthaltung der übrigen Parameter das Halbieren der Gitterweite konsequenterweise zu einer Reduktion der Berechnungsgeschwindigkeit des expliziten Eulerverfahrens um $\sim 75\%$ führt. Gleichzeitig stellt sich der Einfluss auf die Berechnungsgeschwindigkeit des rangadaptiven Eulerverfahrens bemerkenswerterweise als abhängig von der Wahl des Diffusionskoeffizienten dar: Unter $D = 2 \times 10^{-5}$ geht die Halbierung der Gitterweite mit einer Reduktion der Berechnungsgeschwindigkeit um etwa 30% einher, während die Berechnungsgeschwindigkeit unter $D = 4 \times 10^{-5}$ sogar minimal anzusteigen scheint. Damit erweist sich das rangadaptive Eulerverfahren im Hinblick auf Schnelligkeit vor allem in den beiden Konfigurationen mit $h = \frac{1}{256}$ als überlegen.

Berechnungsgeschwindigkeit in it/s		$h = \frac{1}{128}$	$h = \frac{1}{256}$
$D = 2 \times 10^{-5}$	Rangadaptives Eulerverfahren	19.03 ± 0.09	13.72 ± 0.14
	Explizites Eulerverfahren	13.87 ± 0.03	3.18 ± 0.02
$D = 4 \times 10^{-5}$	Rangadaptives Eulerverfahren	25.86 ± 0.55	26.22 ± 0.26
	Explizites Eulerverfahren	13.86 ± 0.06	3.17 ± 0.01

Tabelle 5.9: Gegenüberstellung der durchschnittlichen Berechnungsgeschwindigkeiten des rangadaptiven Eulerverfahrens und des expliziten Eulerverfahrens unter der in Kapitel 5.2 festgelegten Experimentkonfiguration bei Variation des Diffusionskoeffizienten D und der Gitterweite h.

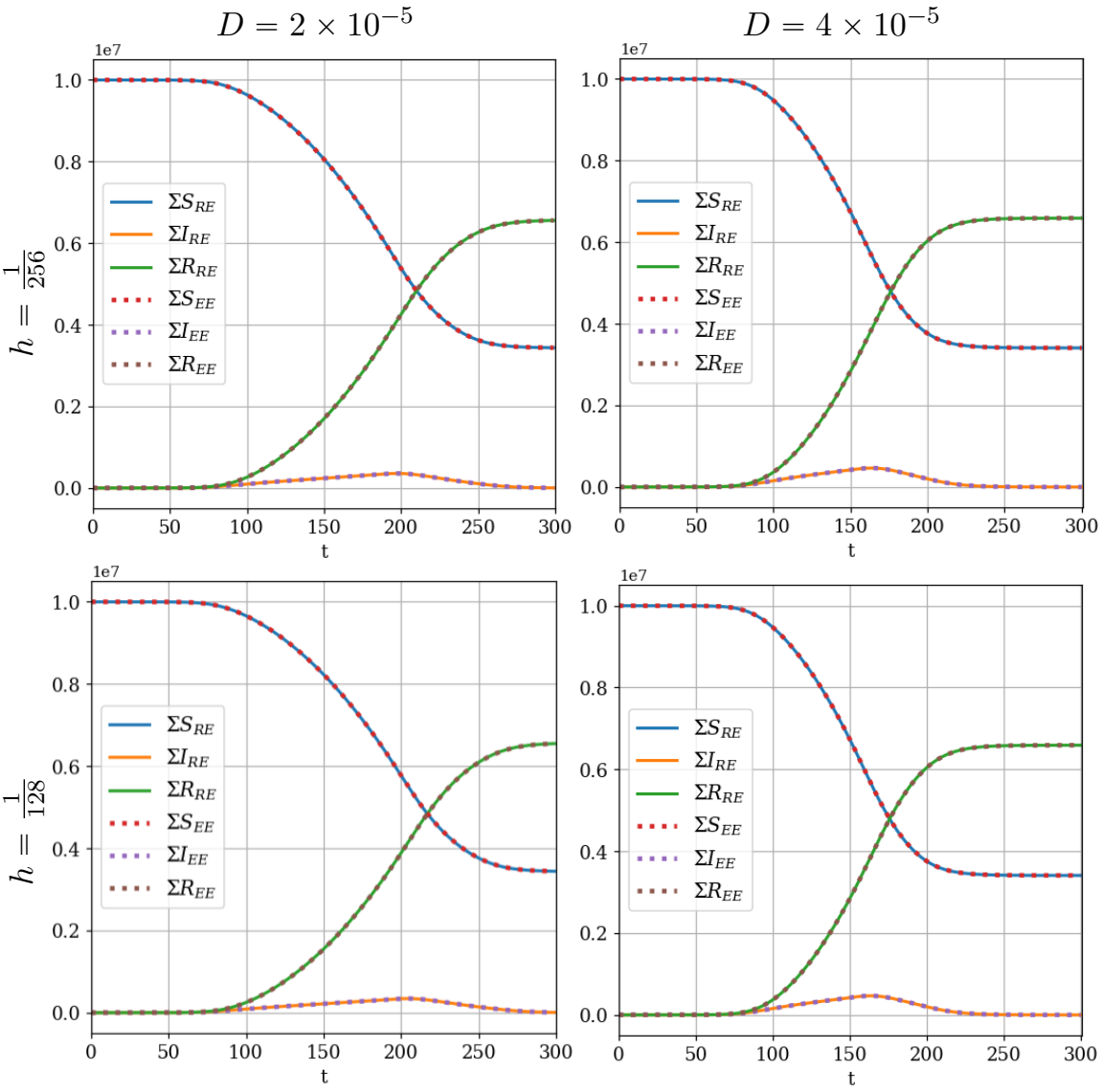


Abbildung 5.10: Vergleich der mit dem rangadaptiven Eulerverfahren (RE) und expliziten Eulerverfahren (EE) berechneten Lösungen des erweiterten SIR-Modells unter der in Kapitel 5.2 gegebenen Experimentkonfiguration bei Variation des Diffusionskoeffizienten D und der Gitterweite h.

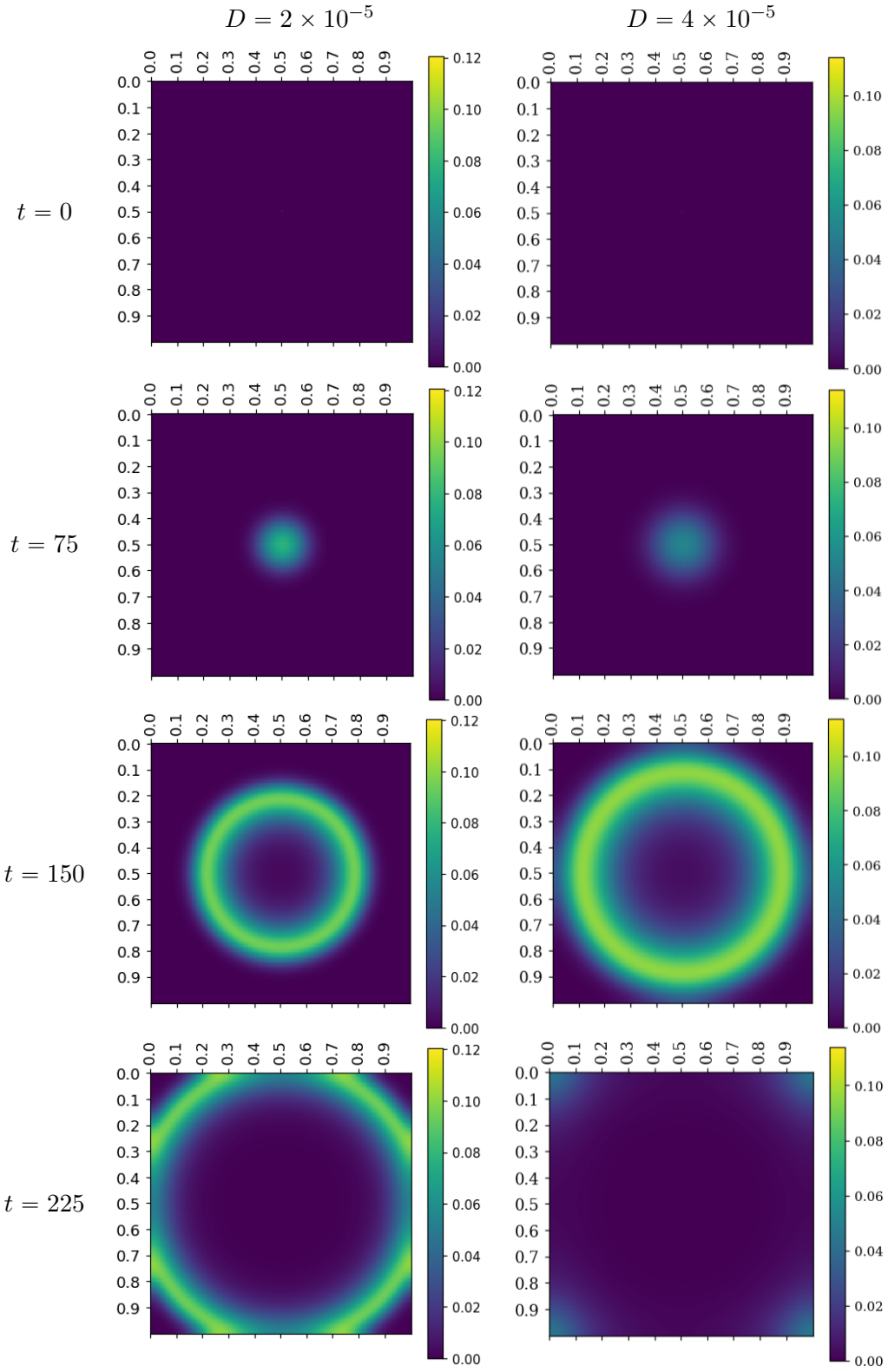


Tabelle 5.10: Vergleich der räumlich aufgelösten relativen Prävalenzen zu vier unterschiedlichen Zeitpunkten für $D = 2 \times 10^{-5}$ und $D = 4 \times 10^{-5}$. Die abgebildeten Lösungen des erweiterten SIR-Modells wurden mit dem rangadaptiven Eulerverfahren unter der in Kapitel 5.2 festgelegten Konfiguration und einer Gitterweite von $h = 1/256$ berechnet.

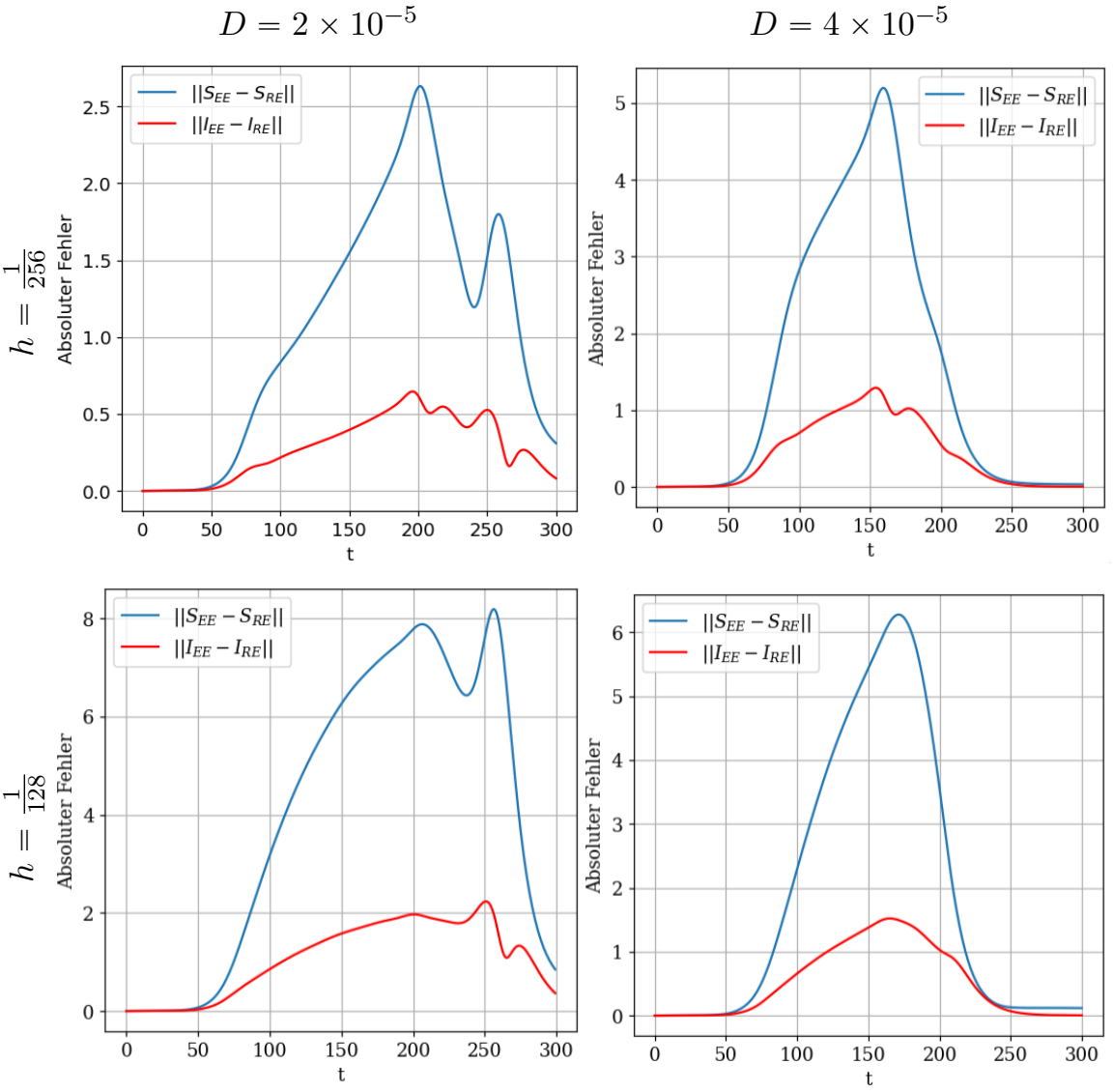


Abbildung 5.11: Abbildung des absoluten Fehlers der auf dem rangadaptiven Eulerverfahren (RE) basierenden Lösung. Als Referenzlösung dient die mit dem expliziten Eulerverfahren (EE) berechnete Lösung. Sämtliche Lösungen beziehen sich auf das erweiterte SIR-Modell unter der in 5.2 definierten Konfiguration bei Variation des Diffusionskoeffizienten D und der Gitterweite h.

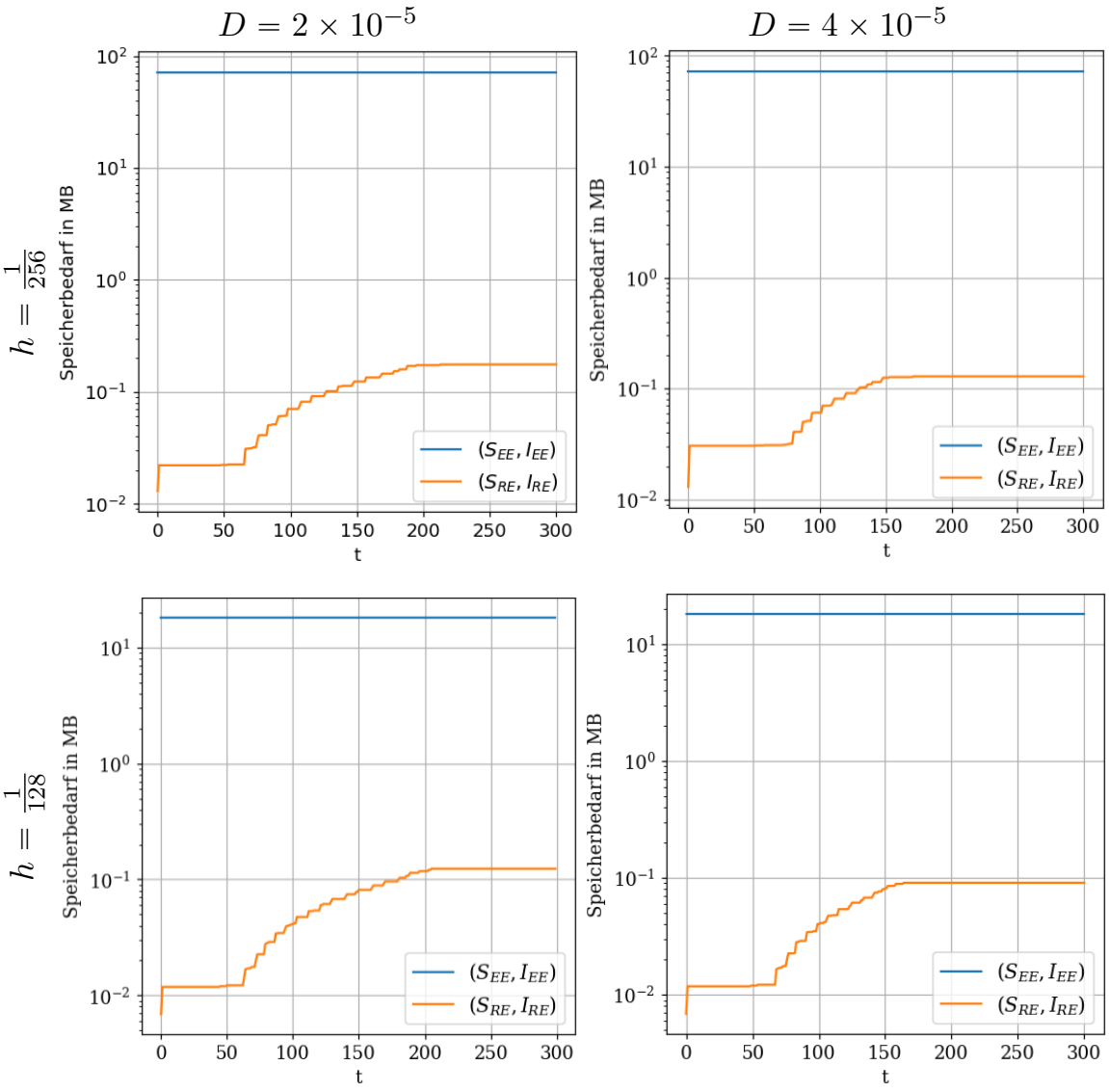


Abbildung 5.12: Die vier Diagramme dieser Abbildung vergleichen den Speicherbedarf der mit dem rangadaptiven Eulerverfahren (RE) und expliziten Eulerverfahren (EE) berechneten Lösungen des erweiterten SIR-Modells unter der in 5.2 festgelegten Konfiguration bei Variation des Diffusionskoeffizienten D und der Gitterweite h.

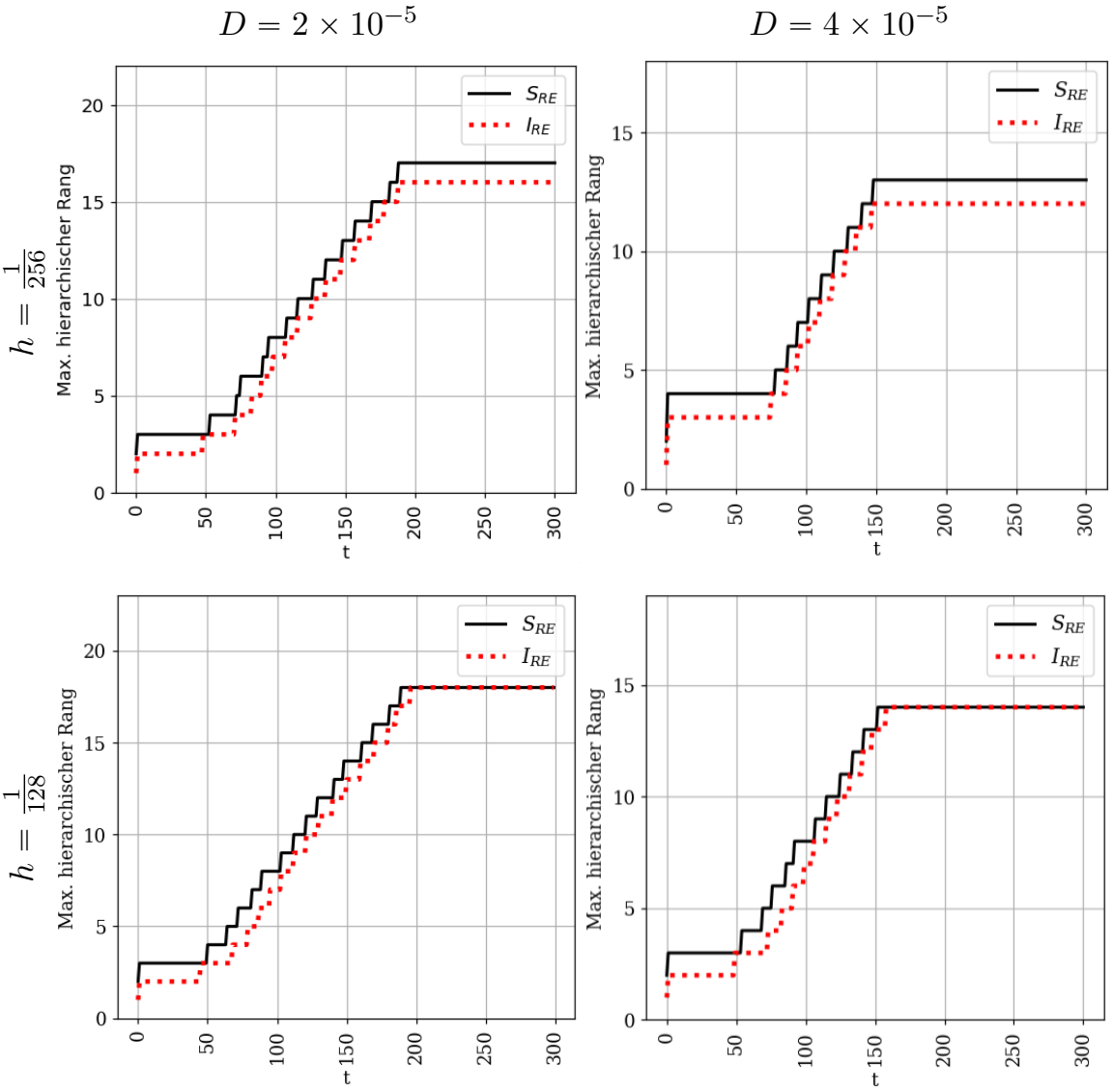


Abbildung 5.13: Diese Abbildung präsentiert den zeitlichen Verlauf der maximalen hierarchischen Ränge der Lösungen des erweiterten SIR-Modells, berechnet mit dem rangadaptiven Eulerverfahren (RE). Die Berechnungen erfolgten unter Verwendung der in Abschnitt 5.2 definierten Konfiguration, wobei sowohl der Diffusionskoeffizient D als auch die Gitterweite h variiert wurden.

6 Fazit

In dieser Masterarbeit wurde anhand eines konkreten mehrdimensionalen Modells untersucht, ob das hierarchische Tuckerformat genutzt werden kann, um die Laufzeit und den Speicherbedarf eines herkömmlichen Lösungsverfahrens zu verbessern. Dazu wurde zunächst die Theorie des hierarchischen Tuckerformats erläutert und für ausgewählte Rechenoperationen gezeigt, wie sich diese im hierarchischen Tuckerformat durchführen lassen. Im Anschluss wurde ein um Altersklassen und Blutgruppen erweitertes SIR-Modell mit Diffusion entwickelt. Darauf aufbauend erfolgte die Verknüpfung beider Themen, indem mit dem rangadaptiven Eulerverfahren ein Lösungsverfahren präsentiert wurde, das die Dynamik des entwickelten Modells im hierarchischen Tuckerformat approximiert. Im abschließenden Teil der Arbeit folgten Experimente, in denen Szenarien unter verschiedenen Parameterkonfigurationen berechnet und diskutiert wurden. Die Berechnungen erfolgten dabei sowohl mit dem rangadaptiven Eulerverfahren und hierarchischen Tuckertensoren als auch mit dem expliziten Eulerverfahren und vollen Tensoren. Das für die Experimente genutzte Python-Framework wurde im Zuge dieser Masterarbeit programmiert und enthält neben einer Implementierung des hierarchischen Tuckerformats auch Jupyter Notebooks zur erneuten Durchführung der Experimente.
Im Rahmen der durchgeführten Experimente zeigte sich der geringe Speicherbedarf des hierarchischen Tuckerformats als wesentlicher Vorteil. Dieser fiel unter jeder Konfiguration zugunsten des hierarchischen Tuckerformats aus, wobei der maximale Unterschied im Vergleich zu den vollen Tensoren bei drei Größenordnungen lag. Der Vergleich der Laufzeiten ergab hingegen ein differenzierteres Bild. Für kleine Problemgrößen war das Verfahren mit vollen Tensoren deutlich überlegen und konnte die Lösung um zwei Größenordnungen schneller berechnen. Da sich die Laufzeit des rangadaptiven Eulerverfahrens mit hierarchischen Tuckertensoren als weniger sensitiv gegenüber Vergrößerungen des Problems erwies, verkehrte sich der ursprüngliche Vorsprung des Verfahrens mit vollen Tensoren bei den größeren Problemen ins Gegenteil. Im besten Fall ergab der Wechsel auf das Verfahren mit hierarchischen Tuckertensoren dabei eine Verachtfachung der Berechnungsgeschwindigkeit. Neben diesen Ergebnissen zur Leistungsfähigkeit des hierarchischen Tuckerformats ließen sich auch einige bemerkenswerte epidemiologische Eigenschaften des Modells herausarbeiten. Es konnte beispielsweise gezeigt werden, dass die altersabhängige Strukturierung der Kontakte einer Population, selbst bei konstanter Gesamtzahl an Kontakten, zu einer Veränderung der Lösung führt, die sich neben einem abgeänderten Gesamtverlauf der Epidemie auch in unterschiedlichen relativen Prävalenzen der verschiedenen Altersklas-

sen bemerkbar macht. Wurde die Ansteckungswahrscheinlichkeit als blutgruppenabhängig gewählt, konnte ein ähnliches Phänomen verzeichnet werden. Des Weiteren konnte das Modell die in der Literatur beschriebene räumliche Ausbreitung der Krankheit in Form einer Wanderwelle erfolgreich reproduzieren.
Ein Nachteil des in dieser Arbeit genutzten rangadaptiven Eulerverfahrens besteht darin, dass es sich um ein explizites Lösungsverfahren handelt, weswegen steife Anfangswertprobleme äußerst kleine Zeitschritte notwendig machen können. Weiterführende Arbeiten könnten sich dieses Problems annehmen und das hierarchische Tuckerformat mit einem impliziten Lösungsverfahren kombinieren. Das epidemiologische Modell selbst bietet ebenfalls einige Möglichkeiten zur Weiterentwicklung. Exemplarisch sei die Ergänzung um eine weitere Dimension genannt, welche es ermöglichen würde, die Individuen der Population auch nach Impfstatus zu unterscheiden. Ferner könnte man die Parameter des Modells als zeit- und ortsabhängig modellieren, um saisonale Effekte und regionale Unterschiedlichkeiten abzubilden.

Literaturverzeichnis

[1] R. Boukhari, A. Breiman, J. Jazat, N. Ruvoën-Clouet, S. Martinez, A. Damais-Cepitelli, C. L. Niger, I. Devie-Hubert, F. Penasse, D. Maurie-re, V. Sébille, A. Dürrbach, and J. L. Pendu, *ABO blood group incompatibility protects against SARS-CoV-2 transmission*, Frontiers in Microbiology, 12 (2022).

[2] P. Gelss, *The Tensor-Train Format and Its Applications*, dissertation, 2017.

[3] L. Grasedyck, *Hierarchical singular value decomposition of tensors*, SIAM Journal on Matrix Analysis and Applications, 31 (2010), pp. 2029–2054.

[4] W. Hackbusch and S. Kühn, *A new scheme for the tensor representation*, Journal of Fourier Analysis and Applications, 15 (2009), pp. 706–722.

[5] W. H. Hamer, *Epidemic disease in england - the evidence of variability and of persi-stence*, The Lancet, 167 (1906), pp. 733–738.

[6] H. W. Hethcote, *Three basic epidemiological models*, in Applied Mathematical Ecology, Springer Berlin Heidelberg, 1989, pp. 119–144.

[7] F. L. Hitchcock, *The expression of a tensor or a polyadic as a sum of products*, Journal of Mathematics and Physics, 6 (1927), pp. 164–189.

[8] Y. HOSONO and B. ILYAS, *TRAVELING WAVES FOR a SIMPLE DIFFUSI-VE EPIDEMIC MODEL*, Mathematical Models and Methods in Applied Sciences, 05 (1995), pp. 935–966.

[9] W. O. Kermack and A. G. McKendrick, *A contribution to the mathematical theory of epidemics*, Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character, 115 (1927), pp. 700–721.

[10] P. Klepac, A. J. Kucharski, A. J. Conlan, S. Kissler, M. L. Tang, H. Fry, and J. R. Gog, *Contacts in context: large-scale setting-specific social mixing matrices from the bbc pandemic project*, medRxiv, (2020).

[11] D. Kressner and C. Tobler, *Algorithm 941*, ACM Transactions on Mathematical Software, 40 (2014), pp. 1–22.

[12] D. R. Kreuz, *Blutgruppenverteilung in der Bevölkerung.* https://www.blutspende.de/magazin/von-a-bis-0/blutgruppen-verteilung-in-der-bevoelkerung. Abgerufen am: 20.10.2023.

[13] L. D. Lathauwer, B. D. Moor, and J. Vandewalle, *A multilinear singular value decomposition*, SIAM Journal on Matrix Analysis and Applications, 21 (2000), pp. 1253–1278.

[14] T. T. Marinov and R. S. Marinova, *Adaptive SIR model with vaccination: simultaneous identification of rates and functions illustrated with COVID-19*, Scientific Reports, 12 (2022).

[15] S. J. Mortensen, L. A. M. Gjerding, M. B. Exsteen, T. Benfield, R. Larsen, F. B. Clausen, K. Rieneck, G. R. Krog, F. Eriksson, and M. H. Dziegiel, *Reduced susceptibility to COVID-19 associated with ABO blood group and pre-existing anti-a and anti-b antibodies*, Immunobiology, 228 (2023), p. 152399.

[16] A. Rodgers, A. Dektor, and D. Venturi, *Adaptive integration of nonlinear evolution equations on tensor manifolds*, Journal of Scientific Computing, 92 (2022).

[17] R. Ross et al., *The prevention of malaria*, J. Murray, London, 1911.

[18] R. Schlickeiser and M. Kröger, *Key epidemic parameters of the sirv model determined from past covid-19 mutant waves*, COVID, 3 (2023), pp. 592–600.

[19] B. Schröter, *Approximation der Dynamik eines epidemiologischen Modells im hierarchischen Tuckerformat.* https://github.com/Beschroe/Masterarbeit, 11 2023.

[20] Statista, *Bevölkerung - Zahl der Einwohner in Deutschland nach relevanten Altersgruppen am 31. Dezember 2022.* https://de.statista.com/statistik/daten/studie/1365/umfrage/bevoelkerung-deutschlands-nach-altersgruppen/. Abgerufen am: 20.10.2023.

[21] A. Sticchi Damiani, A. Zizza, F. Banchelli, M. Gigante, M. L. De Feo, A. Ostuni, V. Marinelli, S. Quagnano, P. Negro, N. Di Renzo, M. Guido, and T. S. C. B. D. S. Group, *Association between abo blood groups and sars-cov-2 infection in blood donors of puglia region*, Annals of Hematology, 102 (2023), pp. 2923–2931.

[22] L. R. Tucker, *Some mathematical notes on three-mode factor analysis*, Psychometrika, 31 (1966), pp. 279–311.

[23] C. Wu, Y. Yang, Q. Zhao, Y. Tian, and Z. Xu, *Epidemic waves of a spatial SIR model in combination with random dispersal and non-local dispersal*, Applied Mathematics and Computation, 313 (2017), pp. 122–143.

A Zusätzliche Definitionen

Definition A.1 (Umgekehrt-lexikographische Ordnung)
Seien $a = (a_1, ..., a_n)$ und $b = (b_1, ..., b_n)$ zwei Tupel der Länge n mit $a_i, b_i \in \mathbb{N} \forall i$, die sich in mindestens einem Eintrag unterscheiden. Außerdem sei j der Index des letzten Eintrags, in dem sich die beiden Tupel unterscheiden. Dann ist a genau dann umgekehrt-lexikographisch kleiner als b, falls $a_j < b_j$.

Definition A.2 (Kronecker-Produkt)
Seien $A \in \mathbb{R}^{n \times m}$ und $B \in \mathbb{R}^{k \times l}$ zwei Matrizen. Dann entspricht das *Kronecker-Produkt* $A \otimes_{\mathcal{K}} B$ folgender $n \cdot k \times m \cdot l$ Matrix :

$$A \otimes_{\mathcal{K}} B = \begin{bmatrix} a_{11} \cdot B & \dots & a_{1m} \cdot B \\ \vdots & \ddots & \vdots \\ a_{n1} \cdot B & \dots & a_{nm} \cdot B \end{bmatrix}$$

bzw.

$$A \otimes_{\mathcal{K}} B = \begin{bmatrix} a_{11} \cdot b_{11} & \dots & a_{11} \cdot b_{1l} & \dots & \dots & a_{1m} b_{11} & \dots & a_{1m} \cdot b_{1l} \\ \vdots & \ddots & \vdots & & & \vdots & \ddots & \vdots \\ a_{11} \cdot b_{k1} & \dots & a_{11} \cdot b_{kl} & \dots & \dots & a_{1m} b_{k1} & \dots & a_{1m} \cdot b_{kl} \\ \vdots & & \vdots & \ddots & & \vdots & & \vdots \\ \vdots & & \vdots & & \ddots & \vdots & & \vdots \\ a_{n1} \cdot b_{11} & \dots & a_{n1} \cdot b_{1l} & \dots & \dots & a_{nm} b_{11} & \dots & a_{nm} \cdot b_{1l} \\ \vdots & \ddots & \vdots & & & \vdots & \ddots & \vdots \\ a_{n1} \cdot b_{k1} & \dots & a_{n1} \cdot b_{kl} & \dots & \dots & a_{nm} b_{k1} & \dots & a_{nm} \cdot b_{kl} \end{bmatrix}$$

Das Kronecker-Produkt der Matrizen A und B enthält damit jedes mögliche Produkt aus zwei Einträgen von A und B.
Handelt es sich bei $A \in \mathbb{R}^n$ und $B \in \mathbb{R}^m$ um zwei Vektoren, wird das Kronecker-Produkt $A \otimes_{\mathcal{K}} wB \in \mathbb{R}^{n \cdot m}$ berechnet, indem implizit angenommen wird, dass A und B Matrizen mit nur einer Spalte sind.

Definition A.3 (Khatri-Rao-Produkt)
Seien $X \in \mathbb{R}^{n \times m}$ und $Y \in \mathbb{R}^{k \times m}$ zwei Matrizen mit

$$X = \left[X_1 | \dots | X_m \right] \text{ und } Y = \left[Y_1 | \dots | Y_m \right].$$

Dann bezeichnet $X \odot Y$ das *Khatri-Rao-Produkt* von X und Y und ist wie folgt definiert:

$$X \odot Y := \left[X_1 \otimes_{\mathcal{K}} Y_1 | \dots | X_m \otimes_{\mathcal{K}} Y_m \right] \in \mathbb{R}^{nk \times m}$$

Der Ausdruck $X \odot^T Y$ notiert wiederum das *transponierte Khatri-Rao-Produkt* von X und Y und ist durch

$$X \odot^T Y := (X^T \odot Y^T)^T \in \mathbb{R}^{m \times nk}$$

bestimmt.

Definition A.4 (Kronecker-Produkt für Tensoren)
Seien $X \in \mathbb{R}^{n_1 \times \dots \times n_d}$ und $Y \in \mathbb{R}^{m_1 \times \dots \times m_d}$ zwei Tensoren der Ordnung d. Dann bezeichnet $X \otimes_{\mathcal{K}} Y$ das Kronecker-Produkt beider Tensoren gemäß folgender Definition:

$$\begin{aligned} &X \otimes_{\mathcal{K}} Y \in \mathbb{R}^{n_1 m_1 \times \dots \times n_d m_d} \\ &(X \otimes_{\mathcal{K}} Y)[k_1, ..., k_d] := X[i_1, ..., i_d] Y[j_1, ..., j_d] \\ &\text{mit } k_\mu := (i_\mu - 1) m_\mu + j_\mu \end{aligned}$$

B Beweise

Beweis B.1 (Multilinearität der multilinearen Multiplikation)
Zu zeigen sind Additivität und Homogenität in allen Argumenten. Seien also $A_i \in \mathbb{R}^{m_i \times n_i}$ mit $i \in \{1, ..., d\}$, $B \in \mathbb{R}^{m_1 \times n_1}$ und $X \in \mathbb{R}^{n_1 \times ... \times n_d}$ sowie $a \in \mathbb{R}$

1. Additivität im ersten Argument:

$$
\begin{aligned}
&((A_1 + B, ..., A_d) \circ X)\,[i_1, ..., i_d] = ((A_1 + B) \circ_1 A_2 \circ_2 ... \, A_d \circ_d X))\,[i_1, ..., i_d] \\
&= \sum_{j=1}^{n_1} (A_1 + B)[i_1, j] \cdot (A_2 \circ_2 ... \, A_d \circ_d X)[j, i_2, ..., i_d] \\
&= \sum_{j=1}^{n_1} (A_1[i_1, j] + B[i_1, j]) \cdot (A_2 \circ_2 ... \, A_d \circ_d X)[j, i_2, ..., i_d] \\
&= \sum_{j=1}^{n_1} A_1[i_1, j] \cdot ((A_2, ..., A_d) \circ X)[j, i_2, ..., i_d] + B[i_1, j] \cdot (A_2 \circ_2 ... \, A_d \circ_d X)[j, i_2, ..., i_d] \\
&= \sum_{j=1}^{n_1} A_1[i_1, j] \cdot (A_2 \circ_2 ... \, A_d \circ_d X)[j, i_2, ..., i_d] + \sum_{j=1}^{n_1} B[i_1, j] \cdot (A_2 \circ_2 ... \, A_d \circ_d X)[j, i_2, ..., i_d] \\
&= (A_1 \circ_1 A_2 \circ_2 ... \, A_d \circ_d X)[i_1, ..., i_d] + (B \circ_1 A_2 \circ_2 ... \, A_d \circ_d X)[i_1, ..., i_d] \\
&= ((A_1, ..., A_d) \circ X)\,[i_1, ..., i_d] + ((B, ..., A_d) \circ X)\,[i_1, ..., i_d]
\end{aligned}
$$

2. Homogenität im ersten Argument:

$$
\begin{aligned}
&((a \cdot A_1), ..., A_d) \circ X)[i_1, ..., i_d] = ((a \cdot A_1) \circ_1 A_2 \circ_2 ... \, A_d \circ_d X)[i_1, ..., i_d] \\
&= \sum_{j=1}^{n_1} (a \cdot A_1)[i_1, j] \cdot (A_2 \circ_2 ... \, A_d \circ_d X)[j, i_2, ..., i_d] \\
&= a \cdot \sum_{j=1}^{n_1} A_1[i_1, j] \cdot (A_2 \circ_2 ... \, A_d \circ_d X)[j, i_2, ..., i_d] \\
&= a \cdot (A_1 \circ_1 A_2 \circ_2 ... \, A_d \circ_d X)[i_1, ..., i_d] \\
&= a \cdot ((A_1, ..., A_d) \circ X)[i_1, ..., i_d]
\end{aligned}
$$

3. Additivität und Homogenität können für alle weiteren Argumente analog gezeigt werden.

□

Beweis B.2 (Bilinearität des äußeren Produkts)
Zu zeigen ist Additivität und Homogenität in beiden Argumenten. Seien im Folgenden also $X, Z \in \mathbb{R}^{n_1 \times ... \times n_d}$ und $Y \in \mathbb{R}^{m_1 \times ... \times m_k}$.

1. Additivität im ersten Argument: $(X + Z) \otimes Y = (X \otimes Y) + (Z \otimes Y)$

$$\begin{aligned}((X + Z) \otimes Y)[i_1, ..., i_{d+k}] &= (X + Z)[i_1, ..., i_d] \cdot Y[i_{d+1}, ..., i_{d+k}] \\ &= (X[i_1, ..., i_d] + Z[i_1, ..., i_d]) \cdot Y[i_{d+1}, ..., i_{d+k}] \\ &= X[i_1, ..., i_d] \cdot Y[i_{d+1}, ..., i_{d+k}] + Z[i_1, ..., i_d] \cdot Y[i_{d+1}, ..., i_{d+k}] \\ &= (X \otimes Y)[i_1, ..., i_{d+k}] + (Z \otimes Y)[i_1, ..., i_{d+k}]\end{aligned}$$

2. Homogenität im ersten Argument: $(a \cdot X) \otimes Y = a \cdot (X \otimes Y)$

$$\begin{aligned}((a \cdot X) \otimes Y)[i_1, ..., i_{d+k}] &= (a \cdot X)[i_1, ..., i_d] \cdot Y[i_{d+1}, ..., i_{d+k}] \\ &= a \cdot X[i_1, ..., i_d] \cdot Y[i_{d+1}, ..., i_{d+k}] \\ &= a \cdot (X \otimes Y)[i_1, ..., i_{d+k}]\end{aligned}$$

3. Additivität und Homogenität können für das zweite Argument analog gezeigt werden.

□

Beweis B.3
Es wird gezeigt, dass das rangadaptive Eulerverfahren angewendet auf das tensorwertige Anfangswertproblem des erweiterten SIR-Modells (4.11) die in Theorem 4.2 vorausgesetzten Ungleichungen

$$||\mathcal{N}(f_n) \; - \; \mathfrak{T}_{r_n}(\mathcal{N}(f_n))|| \leqslant M_1 \Delta t \tag{B.1}$$

$$||f_n \; + \; \Delta t \, \mathfrak{T}_{r_n}(\mathcal{N}(f_n)) \; - \; \mathfrak{T}_{k_n}(f_k \; + \; \Delta t \, \mathfrak{T}_{r_n}(\mathcal{N}(f_n)))|| \leqslant M_2 \Delta t^2 \tag{B.2}$$

erfüllt, sofern die hierarchischen Ränge k_n, $\tilde{k}_n$, r_n, $\tilde{r_n}$ und s_n so gewählt sind, dass die zugehörigen Kürzungsoperatoren mit folgenden Fehlerschranken einhergehen:

$$\mathfrak{T}_{k_n} \text{ und } \mathfrak{T}_{\tilde{k}_n} : \frac{M_2 \Delta t^2}{2} \tag{B.3}$$

$$\mathfrak{T}_{r_n}, \; \mathfrak{T}_{\tilde{r}_n} \text{ und } \mathfrak{T}_{s_n} : \frac{M_1 \Delta t}{4} \tag{B.4}$$

Wegen obiger Wahl der hierarchischen Ränge gilt:

$$\begin{aligned}&I) \; || - S* < \lambda, I^n >_{3,4}^{1,2} + D(A_x \circ_3 S^n + A_y \circ_4 S^n) \\ &- \mathfrak{T}_{r_n} \Big(\mathfrak{T}_{s_n}(-S* < \lambda, I^n >_{3,4}^{1,2}) + D(A_x \circ_3 S^n + A_y \circ_4 S^n) \Big) ||\end{aligned} \tag{B.5}$$

$$
\begin{aligned}
&\leqslant || - S* < \lambda, I^n >_{3,4}^{1,2} + D(A_x \circ_3 S^n + A_y \circ_4 S^n) \\
&- (\mathfrak{T}_{s_n}(-S* < \lambda, I^n >_{3,4}^{1,2}) + D(A_x \circ_3 S^n + A_y \circ_4 S^n))|| \\
&+ ||(\mathfrak{T}_{s_n}(-S* < \lambda, I^n >_{3,4}^{1,2}) + D(A_x \circ_3 S^n + A_y \circ_4 S^n) \\
&- \mathfrak{T}_{r_n}\left(\mathfrak{T}_{s_n}(-S* < \lambda, I^n >_{3,4}^{1,2}) + D(A_x \circ_3 S^n + A_y \circ_4 S^n)\right) || \\
&\overset{B.4}{\leqslant} \frac{M1\Delta t}{4} + \frac{M_1\Delta t}{4} = \frac{M_1\Delta t}{2}
\end{aligned}
$$

$$
\begin{aligned}
II)\ &||S* < \lambda, I^n >_{3,4}^{1,2} - \gamma \star_1 I^n + D(A_x \circ_3 I^n + A_y \circ_4 I^n) \\
&- \mathfrak{T}_{\tilde{r}_n}\left(\mathfrak{T}_{s_n}(S* < \lambda, I^n >_{3,4}^{1,2}) - \gamma \star_1 I^n + D(A_x \circ_3 I^n + A_y \circ_4 I^n)\right) || \\
&\leqslant ||S* < \lambda, I^n >_{3,4}^{1,2} - \gamma \star_1 I^n + D(A_x \circ_3 I^n + A_y \circ_4 I^n) \\
&- (\mathfrak{T}_{s_n}(S* < \lambda, I^n >_{3,4}^{1,2}) - \gamma \star_1 I^n + D(A_x \circ_3 I^n + A_y \circ_4 I^n))|| \\
&+ ||(\mathfrak{T}_{s_n}(S* < \lambda, I^n >_{3,4}^{1,2}) - \gamma \star_1 I^n + D(A_x \circ_3 I^n + A_y \circ_4 I^n) \\
&- \mathfrak{T}_{\tilde{r}_n}\left(\mathfrak{T}_{s_n}(S* < \lambda, I^n >_{3,4}^{1,2}) - \gamma \star_1 I^n + D(A_x \circ_3 I^n + A_y \circ_4 I^n)\right) || \\
&\overset{B.4}{\leqslant} \frac{M1\Delta t}{4} + \frac{M_1\Delta t}{4} = \frac{M_1\Delta t}{2}
\end{aligned} \tag{B.6}
$$

$$
\begin{aligned}
III)\ &||S^n + \Delta t\, \mathfrak{T}_{r_n}\left(\mathfrak{T}_{s_n}(-S* < \lambda, I^n >_{3,4}^{1,2}) + D(A_x \circ_3 S^n + A_y \circ_4 S^n)\right) \\
&- \mathfrak{T}_{k_n}\left(S^n + \Delta t\, \mathfrak{T}_{r_n}\left(\mathfrak{T}_{s_n}(-S* < \lambda, I^n >_{3,4}^{1,2}) + D(A_x \circ_3 S^n + A_y \circ_4 S^n)\right)\right) || \\
&\overset{B.3}{\leqslant} \frac{M_2\Delta t^2}{2}
\end{aligned} \tag{B.7}
$$

$$
\begin{aligned}
IV)\ &||I^n + \Delta t\, \mathfrak{T}_{\tilde{r}_n}\left(\mathfrak{T}_{s_n}(S* < \lambda, I^n >_{3,4}^{1,2}) - \gamma \star_1 I^n + D(A_x \circ_3 I^n + A_y \circ_4 I^n)\right) \\
&- \mathfrak{T}_{\tilde{k}_n}\left(I^n + \Delta t\, \mathfrak{T}_{\tilde{r}_n}\left(\mathfrak{T}_{s_n}(S* < \lambda, I^n >_{3,4}^{1,2}) - \gamma \star_1 I^n + D(A_x \circ_3 I^n + A_y \circ_4 I^n)\right)\right) || \\
&\overset{B.3}{\leqslant} \frac{M_2\Delta t^2}{2}
\end{aligned} \tag{B.8}
$$

Damit ist Ungleichung (B.2) erfüllt, denn es gilt:

$$
\begin{aligned}
&\left|\left|\begin{bmatrix} S^n + \Delta t\, \mathfrak{T}_{r_n}\left(\mathfrak{T}_{s_n}(-S* < \lambda, I^n >_{3,4}^{1,2}) + D(A_x \circ_3 S^n + A_y \circ_4 S^n)\right) \\ I^n + \Delta t\, \mathfrak{T}_{\tilde{r}_n}\left(\mathfrak{T}_{s_n}(S* < \lambda, I^n >_{3,4}^{1,2}) - \gamma \star_1 I^n + D(A_x \circ_3 I^n + A_y \circ_4 I^n)\right) \end{bmatrix}\right.\right. \\
&\left.\left. - \begin{bmatrix} \mathfrak{T}_{k_n}\left(S^n + \Delta t\, \mathfrak{T}_{r_n}\left(\mathfrak{T}_{s_n}(-S* < \lambda, I^n >_{3,4}^{1,2}) + D(A_x \circ_3 S^n + A_y \circ_4 S^n)\right)\right) \\ -\mathfrak{T}_{\tilde{k}_n}\left(I^n + \Delta t\, \mathfrak{T}_{\tilde{r}_n}\left(\mathfrak{T}_{s_n}(S* < \lambda, I^n >_{3,4}^{1,2}) - \gamma \star_1 I^n + D(A_x \circ_3 I^n + A_y \circ_4 I^n)\right)\right) \end{bmatrix}\right|\right| \\
&\leqslant ||S^n + \Delta t\, \mathfrak{T}_{r_n}\left(\mathfrak{T}_{s_n}(-S* < \lambda, I^n >_{3,4}^{1,2}) + D(A_x \circ_3 S^n + A_y \circ_4 S^n)\right) \\
&- \mathfrak{T}_{k_n}\left(S^n + \Delta t\, \mathfrak{T}_{r_n}\left(\mathfrak{T}_{s_n}(-S* < \lambda, I^n >_{3,4}^{1,2}) + D(A_x \circ_3 S^n + A_y \circ_4 S^n)\right)\right) || \\
&+ ||I^n + \Delta t\, \mathfrak{T}_{\tilde{r}_n}\left(\mathfrak{T}_{s_n}(S* < \lambda, I^n >_{3,4}^{1,2}) - \gamma \star_1 I^n + D(A_x \circ_3 I^n + A_y \circ_4 I^n)\right)
\end{aligned}
$$

$$
\begin{aligned}
&- \mathfrak{T}_{\tilde{k}_n}\left(I^n + \Delta t\, \mathfrak{T}_{\tilde{r}_n}\left(\mathfrak{T}_{s_n}(S* < \lambda, I^n >_{3,4}^{1,2}) - \gamma \star_1 I^n + D(A_x \circ_3 I^n + A_y \circ_4 I^n)\right)\right) || \\
&\overset{B.7,B.8}{\leqslant} \frac{M_2\Delta t^2}{2} + \frac{M_2\Delta t^2}{2} = M_2\Delta t^2
\end{aligned}
$$

Ebenfalls ist Ungleichung (B.1) erfüllt, wie folgende Rechnung zeigt:

$$
\begin{aligned}
&\left|\left|\begin{bmatrix} -S* < \lambda, I^n >_{3,4}^{1,2} + D(A_x \circ_3 S^n + A_y \circ_4 S^n) \\ S* < \lambda, I^n >_{3,4}^{1,2} - \gamma \star_1 I^n + D(A_x \circ_3 I^n + A_y \circ_4 I^n) \end{bmatrix}\right.\right. \\
&\left.\left. - \begin{bmatrix} \mathfrak{T}_{r_n}\left(\mathfrak{T}_{s_n}(-S* < \lambda, I^n >_{3,4}^{1,2}) + D(A_x \circ_3 S^n + A_y \circ_4 S^n)\right) \\ \mathfrak{T}_{\tilde{r}_n}\left(\mathfrak{T}_{s_n}(S* < \lambda, I^n >_{3,4}^{1,2}) - \gamma \star_1 I^n + D(A_x \circ_3 I^n + A_y \circ_4 I^n)\right) \end{bmatrix}\right|\right| \\
&\leqslant || - S* < \lambda, I^n >_{3,4}^{1,2} + D(A_x \circ_3 S^n + A_y \circ_4 S^n) \\
&- \mathfrak{T}_{r_n}\left(\mathfrak{T}_{s_n}(-S* < \lambda, I^n >_{3,4}^{1,2}) + D(A_x \circ_3 S^n + A_y \circ_4 S^n)\right) || \\
&+ ||S* < \lambda, I^n >_{3,4}^{1,2} - \gamma \star_1 I^n + D(A_x \circ_3 I^n + A_y \circ_4 I^n) \\
&- \mathfrak{T}_{\tilde{r}_n}\left(\mathfrak{T}_{s_n}(S* < \lambda, I^n >_{3,4}^{1,2}) - \gamma \star_1 I^n + D(A_x \circ_3 I^n + A_y \circ_4 I^n)\right) || \\
&\overset{B.5,B.6}{\leqslant} \frac{M_1\Delta t}{2} + \frac{M_1\Delta t}{2} = M_1\Delta t
\end{aligned}
$$

□