

# Effiziente Berechnung von Varianzen in Markow-Ketten

Maximilian Starke

*Fakultät für Informatik, Technische Universität Dresden*

1. Oktober 2020

## Inhaltsverzeichnis

<b>1</b>	<b>Einführung</b>	<b>2</b>
<b>2</b>	<b>Grundlagen</b>	<b>2</b>
2.1	Wahrscheinlichkeitstheorie . . . . .	2
2.2	Markow-Ketten . . . . .	6
<b>3</b>	<b>Formale Herleitung eines Algorithmus</b>	<b>10</b>
3.1	Berechnung von Erwartungswerten . . . . .	10
3.2	Berechnung von Varianzen . . . . .	12
3.3	Berechnung von Kovarianzen . . . . .	15
3.4	Die initiale Verteilung . . . . .	16
3.5	Komplexität . . . . .	17
<b>4</b>	<b>Performancemessung am praktischen Beispiel</b>	<b>18</b>
4.1	Herman's selbst-stabilisierender Algorithmus . . . . .	18
4.2	Implementation . . . . .	20
4.3	Analyse der Messwerte . . . . .	21
4.4	Grenzen der Aussagekraft . . . . .	23
<b>5</b>	<b>Ein Blick auf MDPs</b>	<b>25</b>
5.1	Was sind MDPs? . . . . .	25
5.2	Resultierende Markow-Ketten . . . . .	26
5.3	Ein Ansatz zum Minimieren von Varianzen . . . . .	29
5.4	Minimieren von Target-Varianzen . . . . .	32
<b>6</b>	<b>Zusammenfassung</b>	<b>35</b>
<b>7</b>	<b>Appendix</b>	<b>36</b>

# 1 Einführung

Zur Analyse relevanter Zielgrößen in probabilistischen Systemen sind Markow-Ketten mit Kantengewichten ein wesentliches, häufig genutztes Modell. Mit dessen Hilfe kann beispielsweise die Dauer eines Verbindungsaufbaus in Netzwerken modelliert werden. Wir wollen in dieser Arbeit Varianzen akkumulierter Kantengewichte auf endlichen Pfaden in Markow-Ketten bis zum Erreichen einer Menge von Zielzuständen und deren effiziente Berechnung betrachten. Während Erwartungswerte der akkumulierten Kantengewichte in der wissenschaftlichen Literatur bereits ausführlich untersucht worden sind, trifft dies nicht auf die Varianzen zu, obgleich diese besonders in sicherheitskritischen Systemen oder zur Beurteilung von Risiken durch Abweichung von der Erwartung besondere Relevanz erhalten. Verhoeff [8] trug praktische Anwendungen zusammen, von denen wir hier zwei erläutern wollen:

Die niederländische Regierung entschied einst in Bezug auf Probleme mit hohem Verkehrsaufkommen, anstatt auf möglichst geringe Erwartungswerte der Fahrzeit hinzuwirken, eher die Minimierung der Varianz der Fahrzeit in den Fokus zu nehmen. Auf diese Weise ist man im Durchschnitt zwar länger unterwegs, erreicht aber mehr Planungssicherheit, kann also Ankunftszeiten genauer vorhersagen: Wenn wir mit einer gewissen Wahrscheinlichkeit  $\geq p$  rechtzeitig an einem Ort ankommen möchten, dann subtrahieren wir von der gewünschten Ankunftszeit den Erwartungswert der Fahrdauer und einen Zeitpuffer, welcher wesentlich von der Varianz abhängt.

In der Wirtschaft kommt es immer wieder vor, dass Budgets für bestimmte Ausgaben geplant werden. Insofern größere Abweichungen der Kosten nach oben unbedingt zu vermeiden sind, kann es von Vorteil sein, anstatt eine Investition in Höhe von fiktiven 1000€ zu planen, bei der mit 300€ Abweichung der tatsächlichen Kosten gerechnet werden muss, eine alternative Investition zum selben Zweck in Höhe von erwarteten 1100€ anzustreben, bei welcher eine Abweichung von nur 50€ erwartet wird.

Verhoeff präsentierte lineare Gleichungssysteme für die Berechnung von Varianzen und Kovarianzen [8]. Wir werden zunächst die erforderlichen Grundlagen zu Wahrscheinlichkeitstheorie sowie Markow-Ketten für den Leser darlegen. Anschließend werden wir einen Algorithmus zur Berechnung von Varianzen und Kovarianzen akkumulierter Kantengewichte formal herleiten. Wir werden die Betrachtungen von Verhoeff insbesondere um eine ausführliche Herleitung für die Berechnung von Kovarianzen erweitern und zeigen, dass alle ermittelten Gleichungssysteme tatsächlich eindeutig lösbar sind. Danach werden wir eine Implementation des hergeleiteten Algorithmus hinsichtlich ihrer Performance analysieren. Zum Abschluss werden wir noch einmal den Blick auf Markow-Entscheidungsprozesse lenken und uns der Problemstellung widmen, wie Varianzen minimiert werden können.

## 2 Grundlagen

### 2.1 Wahrscheinlichkeitstheorie

**Definition 2.1 (diskreter Wahrscheinlichkeitsraum)** Wir nennen ein Paar  $(\Omega, P)$  einen diskreten Wahrscheinlichkeitsraum, wenn  $\Omega$  eine abzählbare Menge

ist und  $P : \Omega \rightarrow [0, 1]$  eine Funktion mit

$$\sum_{\omega \in \Omega} P(\omega) = 1. \quad (1)$$

Wir nennen  $\Omega$  in diesem Zusammenhang auch Ergebnismenge und  $P$  eine Wahrscheinlichkeitsverteilung auf  $\Omega$ .

**Definition 2.2 (Zufallsvariable)** Sei  $(\Omega, P)$  ein diskreter Wahrscheinlichkeitsraum. Eine Funktion  $X : \Omega \rightarrow \mathbb{R}$  heißt Zufallsvariable auf  $(\Omega, P)$ . Wir nennen  $X$  auch kurz Zufallsvariable, falls der Kontext  $(\Omega, P)$  klar ist.

**Definition 2.3 (Erwartungswert)** Sei  $(\Omega, P)$  ein diskreter Wahrscheinlichkeitsraum und  $X$  eine Zufallsvariable auf  $(\Omega, P)$ . Mit

$$\mathcal{E}_{(\Omega, P)}(X) := \sum_{\omega \in \Omega} P(\omega) \cdot X(\omega) \quad (2)$$

bezeichnen wir den Erwartungswert von  $X$  auf  $(\Omega, P)$ . Sollte der Kontext  $(\Omega, P)$  klar sein, schreiben wir auch kurz  $\mathcal{E}(X)$ . Es sei angemerkt, dass es für  $|\Omega| = \infty$  Fälle gibt, bei denen die Summe nicht beschränkt ist. Wir schreiben dafür  $\mathcal{E}(X) = \infty$  bzw.  $\mathcal{E}(X) = -\infty$  und sagen, dass der Erwartungswert nicht existiert.

**Beispiel 2.4** Seien  $\Omega = \mathbb{N}_{>0}$ ,  $P : \Omega \rightarrow [0, 1] : n \mapsto 2^{-n}$  und  $R : \Omega \rightarrow \mathbb{R} : n \mapsto 2^n$ . Dann gibt es offensichtlich für jede Zahl  $a \in \mathbb{R}$  eine endliche Teilmenge von  $T \subseteq \Omega$  mit  $\sum_{n \in T} P(n) \cdot R(n) \geq a$ . Dazu wählen wir einfach  $T := \{n \in \mathbb{N}_{>0} \mid n \leq \lceil a \rceil\}$  und erhalten  $\sum_{n \in T} P(n) \cdot R(n) = |T| = \lceil a \rceil \leq \mathcal{E}(R)$ . Damit erhalten wir  $\mathcal{E}(R) = \infty$ .

Wir werden uns im Folgenden nur auf Zufallsvariablen beziehen, für welche ein Erwartungswert existiert.

**Beispiel 2.5** Ein manipulierter Spielwürfel habe 6 Seiten  $\Omega = \{1, 2, 3, 4, 5, 6\}$ , wobei die 1 im Vergleich zu einem herkömmlichen Würfel mit einer 2 überklebt wurde. Dies drücken wir durch die Zufallsvariable  $X : \Omega \rightarrow \mathbb{R} : s \mapsto \max(2, s)$  aus. Außerdem wurde der Würfel mit ungleichmäßig verteilter Masse so gefertigt, dass nach einem Wurf nicht alle Seiten gleich wahrscheinlich oben liegen. Wir nehmen für dieses Beispiel eine Wahrscheinlichkeitsverteilung mit  $P(1) = 0, 1$ ,  $P(2) = 0, 15$ ,  $P(3) = 0, 15$ ,  $P(4) = 0, 15$ ,  $P(5) = 0, 15$ ,  $P(6) = 0, 3$  an. Dann ist eine Augenzahl pro Wurf von

$$\mathcal{E}(X) = (0, 1 + 0, 15) \cdot 2 + 0, 15 \cdot (3 + 4 + 5) + 0, 3 \cdot 6 = 4, 1$$

zu erwarten und damit 0,6 Augen mehr als bei einem ungezinkten Würfel mit Erwartungswert 3,5. Würden wir nun alle beschrifteten Zahlen noch zusätzlich quadrieren, erhielten wir einen Würfel mit  $Y : \Omega \rightarrow \mathbb{R} : s \mapsto \max(4, s^2)$  und würden im Schnitt 19,3 Augen bei einem Wurf erwarten:

$$\mathcal{E}(Y) = (0, 25) \cdot 4 + 0, 15 \cdot (9 + 16 + 25) + 0, 3 \cdot 36 = 19, 3$$

Es lässt sich leicht die folgende Linearität des Erwartungswertes beobachten:

**Lemma 2.6** Seien  $X, Y$  Zufallsvariablen auf demselben diskreten Wahrscheinlichkeitsraum,  $c, d \in \mathbb{R}$ . Dann gilt:

$$\mathcal{E}(X + cY + d) = \mathcal{E}(X) + c\mathcal{E}(Y) + d \quad (3)$$

Dabei ist  $X + cY + d$  die übliche Notation für die Zufallsvariable gegeben durch  $Z : \Omega \rightarrow \mathbb{R} : \omega \mapsto X(\omega) + c \cdot Y(\omega) + d$ .

*Beweis.*

$$\begin{aligned} \mathcal{E}(X + cY + d) &= \sum_{\omega \in \Omega} P(\omega) \cdot (X + cY + d)(\omega) \\ &= \sum_{\omega \in \Omega} P(\omega) \cdot X(\omega) + c \cdot P(\omega) \cdot Y(\omega) + d \cdot P(\omega) \\ &= \mathcal{E}(X) + c\mathcal{E}(Y) + d \end{aligned} \quad \square$$

**Definition 2.7 (Varianz)** Sei  $(\Omega, P)$  ein diskreter Wahrscheinlichkeitsraum und  $X$  eine Zufallsvariable auf  $(\Omega, P)$ . Mit

$$\mathcal{V}ar_{(\Omega, P)}(X) := \mathcal{E}_{(\Omega, P)} \left( (X - \mathcal{E}_{(\Omega, P)}(X))^2 \right) \quad (4)$$

bezeichnen wir die Varianz von  $X$  auf  $(\Omega, P)$ . Sollte der Kontext  $(\Omega, P)$  klar sein, schreiben wir kurz  $\mathcal{V}ar(X)$ .

**Definition 2.8 (Kovarianz)** Seien  $(\Omega, P)$  ein diskreter Wahrscheinlichkeitsraum und  $X, Y$  Zufallsvariablen auf  $(\Omega, P)$ . Die Kovarianz  $\mathcal{C}ov_{(\Omega, P)}(X, Y)$  ist definiert durch:

$$\mathcal{C}ov_{(\Omega, P)}(X, Y) := \mathcal{E} \left( (X - \mathcal{E}(X))(Y - \mathcal{E}(Y)) \right) \quad (5)$$

Offensichtlich ist die Varianz ein Spezialfall der Kovarianz, denn aus den Definitionen folgt unmittelbar  $\mathcal{V}ar_{(\Omega, P)}(X) = \mathcal{C}ov_{(\Omega, P)}(X, X)$ . Wir betrachten im Folgenden, wie sich Kovarianzen durch Erwartungswerte beschreiben lassen.

**Lemma 2.9** Seien  $X$  und  $Y$  Zufallsvariablen auf demselben diskreten Wahrscheinlichkeitsraum. Dann gilt:

$$\mathcal{C}ov(X, Y) = \mathcal{E}(XY) - \mathcal{E}(X)\mathcal{E}(Y) \quad (6)$$

*Beweis.*

$$\begin{aligned} \mathcal{C}ov(X, Y) &= \mathcal{E} \left( (X - \mathcal{E}(X))(Y - \mathcal{E}(Y)) \right) && \text{(Definition 2.8)} \\ &= \mathcal{E}(XY - X\mathcal{E}(Y) - Y\mathcal{E}(X) + \mathcal{E}(X)\mathcal{E}(Y)) \\ &= \mathcal{E}(XY) - 2\mathcal{E}(X)\mathcal{E}(Y) + \mathcal{E}(X)\mathcal{E}(Y) && \text{(Lemma 2.6)} \\ &= \mathcal{E}(XY) - \mathcal{E}(X)\mathcal{E}(Y) \end{aligned}$$

$\square$

Betrachten wir den Spezialfall  $X = Y$  von Lemma 2.9, dann erhalten wir folgende Beziehung für Varianzen:

**Korollar 2.10** Sei  $X$  eine Zufallsvariable auf einem diskreter Wahrscheinlichkeitsraum. Dann gilt:

$$\mathcal{V}ar(X) = \mathcal{E}(X^2) - \mathcal{E}(X)^2 \quad (7)$$

Ähnlich zur Linearität von Erwartungswerten (3) lässt sich für Varianzen folgende Beziehung feststellen:

**Lemma 2.11** Sei  $X$  eine Zufallsvariable auf einem diskreter Wahrscheinlichkeitsraum,  $c, d \in \mathbb{R}$ . Dann gilt:

$$\mathcal{V}ar(cX + d) = c^2 \cdot \mathcal{V}ar(X) \quad (8)$$

*Beweis.*

$$\begin{aligned} \mathcal{V}ar(cX + d) &= \mathcal{E}\left((cX + d - \mathcal{E}(cX + d))^2\right) && \text{(Definition 2.7)} \\ &= \mathcal{E}\left((cX + d - c\mathcal{E}(X) - d)^2\right) && \text{(Lemma 2.6)} \\ &= \mathcal{E}\left(c^2 \cdot (X - \mathcal{E}(X))^2\right) \\ &= c^2 \cdot \mathcal{E}\left((X - \mathcal{E}(X))^2\right) && \text{(Lemma 2.6)} \\ &= c^2 \cdot \mathcal{V}ar(X) && \text{(Definition 2.7)} \quad \square \end{aligned}$$

Entsprechend der Intuition ist die Varianz, das Quadrat der Standardabweichung, invariant unter Addition einer Konstanten zur Zufallsvariable. Wird jedoch die Zufallsvariable mit einem Faktor skaliert, so ändert sich die Varianz um das Quadrat des Faktors. Aus gutem Grund wird die Varianz auch mittlere quadratische Abweichung genannt.

Durch das Wegfallen von  $d$  lässt sich recht trivial auf Gleichungen schließen, welche unter Umständen nicht mehr auf dem ersten Blick als klar und offensichtlich angesehen werden können. Das folgende Korollar soll dies verdeutlichen:

**Korollar 2.12** Sei  $X$  eine Zufallsvariable und  $c \in \mathbb{R}$ . Dann gilt:

$$\mathcal{E}((c + X)^2) = (c + \mathcal{E}(X))^2 + \mathcal{V}ar(X) \quad (9)$$

*Beweis.* Aus Korollar 2.10 folgt unmittelbar

$$\mathcal{V}ar(X + c) = \mathcal{E}((X + c)^2) - (\mathcal{E}(X + c))^2.$$

Die linke Seite kann nach Lemma 2.11 vereinfacht werden zu  $\mathcal{V}ar(X + c) = \mathcal{V}ar(X)$ . Der Subtrahend  $(\mathcal{E}(X + c))^2$  lässt sich aufgrund der Linearität (Lemma 2.6) auch als  $(\mathcal{E}(X) + c)^2$  schreiben.  $\square$

**Beispiel 2.13** Wir wollen nun Beispiel 2.5 fortsetzen. Nach Korollar 2.10 beträgt die Varianz beim ursprünglichen Würfel also  $\mathcal{V}ar(X) = \mathcal{E}(X^2) - \mathcal{E}(X)^2 = \mathcal{E}(Y) - \mathcal{E}(X)^2 = 19,3 - 16,81 = 2,49$ . Es sei bemerkt, dass wir  $Y$  geschickt so gewählt haben, dass  $Y = X^2$  gilt. Und tatsächlich ergibt die Berechnung nach Definition 2.7 denselben Wert:

$$\begin{aligned} \mathcal{V}ar(X) &= 0,15 \cdot ((3 - 4,1)^2 + (4 - 4,1)^2 + (5 - 4,1)^2) + \\ &\quad + 0,25 \cdot (2 - 4,1)^2 + 0,3 \cdot (6 - 4,1)^2 \\ &= 0,25 \cdot 4,41 + 0,15 \cdot (1,21 + 0,01 + 0,81) + 0,3 \cdot 3,61 \\ &= 2,49 \end{aligned}$$

Die Kovarianz lässt sich nach Lemma 2.9 berechnen als  $\text{Cov}(X, Y) = \mathcal{E}(XY) - \mathcal{E}(X)\mathcal{E}(Y)$ . Wir kennen bereits  $\mathcal{E}(X)$  sowie  $\mathcal{E}(Y)$  und es gilt:

$$\mathcal{E}(XY) = \mathcal{E}(X^3) = 0,25 \cdot 2^3 + 0,15 \cdot (3^3 + 4^3 + 5^3) + 0,3 \cdot 6^3 = 99,2$$

Dann ist  $\text{Cov}(X, Y) = 99,2 - 4,1 \cdot 19,3 = 20,07$  und tatsächlich ergibt die explizite Rechnung denselben Wert:

$$\begin{aligned} \text{Cov}(X, Y) &= \mathcal{E}\left((X - \mathcal{E}(X))(Y - \mathcal{E}(Y))\right) \\ &= 0,25 \cdot (2 - 4,1)(4 - 19,3) \\ &\quad + 0,15 \cdot (3 - 4,1)(9 - 19,3) + 0,15 \cdot (4 - 4,1)(16 - 19,3) \\ &\quad + 0,15 \cdot (5 - 4,1)(25 - 19,3) + 0,3 \cdot (6 - 4,1)(36 - 19,3) \\ &= 20,07 \end{aligned}$$

## 2.2 Markow-Ketten

**Definition 2.14 (Markow-Kette)** Eine Markow-Kette ist ein Tupel  $(Q, P, I)$  mit den Eigenschaften

- (a)  $Q$  ist eine abzählbare Menge.
- (b)  $P : Q \times Q \rightarrow [0, 1]$  mit der Eigenschaft  $\forall q \in Q : \sum_{q' \in Q} P(q, q') = 1$ .
- (c)  $I : Q \rightarrow [0, 1]$  ist eine Wahrscheinlichkeitsverteilung auf  $Q$ .

Die Bilder von  $P$  nennen wir auch Transitionswahrscheinlichkeiten und definieren mit welcher Wahrscheinlichkeit, nämlich  $P(q, q')$  vom aktuellen Zustand  $q$  in den Zustand  $q'$  übergegangen wird.  $I$  ist die initiale Verteilung, welche definiert, mit welcher Wahrscheinlichkeit ein Zustand als Startzustand gewählt wird.

Für theoretische Betrachtungen ist es durchaus sinnvoll unendliche Markow-Ketten zu betrachten. Wir werden uns jedoch auf endliche beschränken, d.h.  $|Q| < \infty$ , sobald wir zur Berechnung von u.a. Varianzen entsprechende Gleichungssysteme herleiten. Andernfalls bestünde das Gleichungssystem aus unendlich vielen Gleichungen in unendlich vielen Variablen.

**Definition 2.15 (Pfad)** Sei  $M = (Q, P, I)$  eine Markow-Kette. Die Menge aller endlichen Pfade in  $M$  sei definiert durch

$$\text{Paths}(M) := \{p \in Q^k \mid k \in \mathbb{N}_{>0} \wedge \forall 0 \leq i < k : P(p_i, p_{i+1}) > 0\} \quad (10)$$

Wir bezeichnen mit  $|p|$  die Größe des Tupels  $p$ , d.h.  $|p| = k$  für  $p \in Q^k$ . Wir beginnen Indizes bei 0. Die Wahrscheinlichkeit  $\tilde{P}(p)$  eines Pfades ist gegeben durch

$$\tilde{P}(p) := \prod_{i=0}^{|p|-2} P(p_i, p_{i+1}) \quad (11)$$

Insbesondere ist  $\tilde{P}(p) = 1$  für alle Pfade  $p$  mit  $|p| = 1$ . Für Tupel  $p \in Q^k$ ,  $k \in \mathbb{N}_{>0}$ , die keine Pfade sind, gilt entsprechend  $\tilde{P}(p) = 0$ .

Für  $|p| = 2$  entspricht die Wahrscheinlichkeit  $\tilde{P}(p)$  genau der Wahrscheinlichkeit gegeben durch die Funktion  $P$ , nämlich der Transitionswahrscheinlichkeit aus der Markow-Kette.  $\tilde{P}$  ergibt sich als eindeutige Fortsetzung von  $P$ , es gilt  $P = \tilde{P}|_{Q \times Q}$ , und wir schreiben daher im Folgenden einfach nur  $P$ .

**Definition 2.16 (Gewichtsfunktion)** Sei  $M = (Q, P, I)$  eine Markow-Kette. Eine Gewichtsfunktion auf  $M$  ist eine Abbildung

$$R : Q \times Q \rightarrow \mathbb{R}. \quad (12)$$

Wir bezeichnen Gewichtsfunktionen typischerweise mit  $R$  in Anlehnung an das englische Wort *reward*. Offenbar gibt es analog zur Wahrscheinlichkeit  $P$  eine eindeutige Fortsetzung für eine Gewichtsfunktion  $R$  auf Pfade, gegeben durch Aufsummierung aller Kantengewichte entlang eines solchen:

$$\tilde{R} : \bigcup_{k \in \mathbb{N}_{>0}} Q^k \rightarrow \mathbb{R} : p \mapsto \sum_{i=0}^{|p|-2} R(p_i, p_{i+1}) \quad (13)$$

Wir schreiben im Folgenden einfach  $R$ .

**Definition 2.17** Sei  $M = (Q, P, I)$  eine Markow-Kette,  $s \in Q$  ein beliebiger Zustand, genannt Startzustand, und  $A \subseteq Q$  eine Menge von Zielzuständen. Wir bezeichnen die Menge der Pfade, welche in  $s$  starten und in  $A$  enden, jedoch  $A$  nicht zwischenzeitlich schon erreichen, mit:

$$\text{Paths}_{s \rightarrow A}(M) := \{p \in \text{Paths}(M) \mid p_0 = s \wedge p_{|p|-1} \in A \wedge \forall i < |p| - 1 : p_i \notin A\} \quad (14)$$

Wir ziehen nun Erkenntnisse heran, die Baier und Katoen in Kapitel 10.1 ihres Buches *Principles of Model Checking* [1] beschreiben, speziell im Abschnitt *Reachability Probabilities*. Setzen wir einmal voraus, dass von jedem Zustand  $q \in Q$ , welcher von  $s$  erreichbar ist, ein Pfad in die Zielzustandsmenge  $A$  existiert, das heißt

$$\text{Paths}_{s \rightarrow \{q\}}(M) \neq \emptyset \quad \Rightarrow \quad \text{Paths}_{q \rightarrow A}(M) \neq \emptyset.$$

Unter dieser Voraussetzung beobachteten Baier und Katoen, dass es ein fast sicheres Ereignis ist, nach endlich vielen Übergängen an einem Zustand aus  $A$  vorbeizukommen. Die endlichen Pfade, welche wir ausschließlich betrachten wollen, lassen sich als Präfixe der unendlichen Pfade  $\text{Paths}_{inf}(s) := \{p \in Q^\omega \mid p(0) = s \wedge \forall i \in \mathbb{N} : P(p_i, p_{i+1}) > 0\}$  der Markow-Kette auffassen. Man sieht leicht, dass sich ein endlicher Pfad mit der Menge an jenen unendlichen Pfaden identifizieren lässt, welche diesen als Präfix haben. Eine Menge endlicher Pfade ist abzählbar, da wir die Pfade als endliche Wörter über dem Alphabet bestehend aus den Zuständen auffassen können. Nach dieser Beobachtung ist dann

$$(\text{Paths}_{s \rightarrow A}(M), P)$$

ein diskreter Wahrscheinlichkeitsraum, da die Menge aller Pfade, welche  $A$  nie besuchen, ein fast unmögliches Ereignis darstellt. Die Fortsetzung von  $R$  ist eine Zufallsvariable auf  $\text{Paths}_{s \rightarrow A}(M)$ .

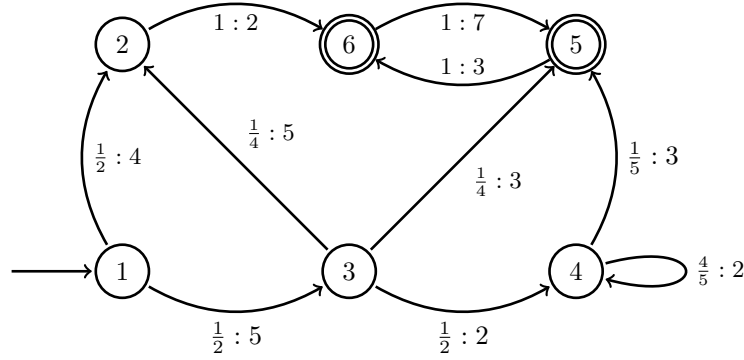
**Beispiel 2.18** Gegeben sei eine Markow-Kette  $M = (Q, P, I)$  mit

$$Q = \{1, 2, 3, 4, 5, 6\}$$

$$P = \begin{pmatrix} 0 & 0,5 & 0,5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0,25 & 0 & 0,5 & 0,25 & 0 \\ 0 & 0 & 0 & 0,8 & 0,2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

$$I = (1 \ 0 \ 0 \ 0 \ 0 \ 0)^\top$$

Wir betrachten also eine Markow-Kette mit dem einzigen initialen Zustand 1. Weiterhin sei  $\{5, 6\}$  die betrachtete Menge an Zielzuständen, welche wir auf Pfaden erreichen wollen. Eine Markow-Kette lässt sich immer auch als Graph auffassen: Wir beschriften alle Kanten  $(i, j)$  mit Übergangswahrscheinlichkeit und Wert der Gewichtsfunktion  $R$  in der Schreibweise  $P(i, j) : R(i, j)$ . Sei die Gewichtsfunktion  $R$  implizit definiert durch folgenden Graphen:



Wir wollen nun zur Veranschaulichung den Erwartungswert und die Varianz von  $R$  im Wahrscheinlichkeitsraum  $(\text{Paths}_{1 \rightarrow \{5,6\}}(M), P)$  betrachten. Die von 5 beziehungsweise 6 ausgehenden Kanten samt Beschriftung sind für diesen Fall irrelevant. Die Tupel aus  $Q^n$  können wir mit Wörtern über dem Alphabet  $Q$  identifizieren. Wir erhalten so:

$$\Omega := \text{Paths}_{1 \rightarrow \{5,6\}}(M) = \{126, 1326, 135\} \cup \{134^n 5 \in Q^{n+3} \mid n \geq 1\}$$

Für den Erwartungswert  $\mathcal{E}(R)$  ergibt sich:

$$\begin{aligned} \mathcal{E}(R) &= \sum_{p \in \Omega} P(p) \cdot R(p) \\ &= \frac{1}{2} \cdot 6_{[126]} + \frac{1}{8} \cdot 12_{[1326]} + \frac{1}{8} \cdot 8_{[135]} + \sum_{n=0}^{\infty} \frac{1}{20} \cdot \left(\frac{4}{5}\right)^n \cdot (10 + 2n)_{[134^n 5]} \\ &= 10 \end{aligned}$$

Für das Berechnen der unendlichen Summe möchten wir an dieser Stelle auf den Appendix verweisen. Mit Korollar 7.2 und 7.4 und der Zerlegung

$$\sum_{n=0}^{\infty} \frac{1}{20} \cdot \left(\frac{4}{5}\right)^n \cdot (10 + 2n) = \frac{1}{2} \sum_{n=0}^{\infty} \left(\frac{4}{5}\right)^n + \frac{1}{10} \sum_{n=0}^{\infty} \left(\frac{4}{5}\right)^n \cdot n$$



lässt sich der Wert dieser Summe, nämlich 4, 5, leicht ausrechnen. Als Varianz  $\text{Var}(R)$  erhalten wir, diesmal unter Verwendung von Korollar 7.6:

$$\begin{aligned}
\text{Var}(R) &= \mathcal{E}((R - 10)^2) \\
&= \frac{1}{2} \cdot (6 - 10)^2 + \frac{1}{8} \cdot (12 - 10)^2 + \frac{1}{8} \cdot (8 - 10)^2 + \\
&\quad + \sum_{n=0}^{\infty} \frac{1}{20} \cdot \left(\frac{4}{5}\right)^n \cdot (10 + 2n - 10)^2 \\
&= 9 + \frac{1}{5} \sum_{n=0}^{\infty} \left(\frac{4}{5}\right)^n \cdot n^2 \\
&= 9 + 36 \\
&= 45
\end{aligned}$$

Es sei angemerkt, dass die Zufallsvariable  $X := (R - 10)^2$ , per Definition

$$(R - 10)^2 : \text{Paths}_{1 \rightarrow \{5,6\}}(M) \rightarrow \mathbb{R} : p \mapsto (R(p) - 10)^2,$$

im Allgemeinen nicht die Eigenschaft der eindeutigen Fortsetzung von Kanten zu Pfaden erfüllt:

$$\forall n \in \mathbb{N}, p \in Q^n : n > 1 \Rightarrow X(p) = \sum_{i=0}^{n-2} X(p(i), p(i+1))$$

Beispielsweise verletzt  $X(1, 3) + X(3, 2) = 25 + 25 \neq 0 = X(132)$  diese Eigenschaft. Während im Graphen eingezeichnete Kantengewichte immer zu einer Zufallsvariable auf der Menge der Pfade fortgesetzt werden können, können wir nicht alle Zufallsvariablen auf Kantengewichte zurückführen, insbesondere  $X$  also nicht durch Kantengewichte im Graphen einzeichnen.

An diesem Beispiel sehen wir, dass die explizite Berechnung von Varianzen bereits kompliziert werden kann, wenn nur ein Kreis trivialer Länge im Graphen der Markow-Kette enthalten ist. Stellen wir uns einen Graphen vor, in dem Knoten  $a$  und  $b$  existieren sowie Kanten  $(a, a)$ ,  $(a, b)$ ,  $(b, a)$ . Dann gibt es nicht mehr nur eine Möglichkeit, einen Kreis zu finden, sondern unendlich viele mit beispielsweise  $(a)^\omega$ ,  $(ab)^\omega$ ,  $(aab)^\omega$ ,  $(aaab)^\omega$ ,  $(aaaab)^\omega$ ,  $\dots$ .

**Definition 2.19** Sei  $M = (Q, P, I)$  eine endliche Markow-Kette,  $A \subseteq Q$  eine Zielmenge. Dann bezeichnen wir mit  $P_{\rightarrow A}$  die folgende Matrix:

$$P_{\rightarrow A} : Q^2 \rightarrow [0, 1] : \begin{cases} P(s, t) & \text{falls } s \notin A \\ 0 & \text{falls } s \in A \end{cases} \quad (15)$$

Diese Modifikation entspricht dem Entfernen von allen ausgehenden Kanten von Zielzuständen  $a \in A$ .

**Satz 2.20** Sei  $M = (Q, P, I)$  eine endliche Markow-Kette,  $A \subseteq Q$  eine Zielmenge, die von jedem Knoten aus erreichbar ist ( $\forall s \in Q : \text{Paths}_{s \rightarrow A}(M) \neq \emptyset$ ). Dann ist die Matrix  $D := P_{\rightarrow A} - \mathbb{1}$  invertierbar.

*Beweis.* Aus der linearen Algebra ist bekannt, dass  $D$  genau dann invertierbar ist, wenn  $Dx = 0$  nur die Lösung  $x = 0$  hat. Nehmen wir also an,  $x \in \mathbb{R}^Q$  mit  $x \neq 0$

erfüllt  $Dx = 0$ . O.B.d.A. habe  $x$  einen positiven Eintrag. (Hat  $x$  keinen positiven Eintrag, so hat es einen negativen Eintrag. Dann finden wir mit  $-x$  eine Lösung mit einem positivem Eintrag, da  $D(-x) = -Dx = 0$ .) Sei  $E \subseteq Q$  die Indexmenge aller maximalen Einträge von  $x$ , also  $e \in E : \Leftrightarrow \forall q \in Q : x_q \leq x_e$ . Sei  $s \in A$ . Dann folgt aus  $(Dx)_s = 0$ , dass  $\sum_{q \in Q} (P_{\rightarrow A} - \mathbb{1})_{(s,q)} \cdot x_q = -\sum_{q \in Q} \mathbb{1}_{(s,q)} \cdot x_q = 0$  und damit  $x_s = 0$ .

Sei nun  $s \in E$  frei gewählt. Dann ist  $x_s > 0$  und daher  $s \notin A$ . Aus  $(Dx)_s = 0$  folgt dann  $\sum_{q \in Q} (P_{\rightarrow A} - \mathbb{1})_{(s,q)} \cdot x_q = 0$ . Damit bekommen wir  $\sum_{q \in Q} P_{(s,q)} \cdot x_q = x_s$ . Da nach Definition 2.14 die Abbildung  $q \mapsto P(s, q)$  eine Wahrscheinlichkeitsverteilung auf  $Q$  ist, folgt schließlich  $\forall q \in Q : (P(s, q) > 0 \Rightarrow x_q = x_s)$ . Schließlich kann die gewichtete Summe nur genau dann den maximalen Wert  $x_s$  annehmen, wenn alle zu gewichtenden Summanden bereits maximal sind. D.h. für jeden Nachfolgeknoten  $t \in Q$  von  $s \in E$  in der Markow-Kette  $M$  liegt  $t$  selbst wieder in  $E$ . Da ein Knoten  $a \in A$  von  $s$  aus erreichbar ist, gilt auch  $a \in E$ . Aber dann wäre  $a \notin A$ . Wir erhalten einen Widerspruch.  $\square$

### 3 Formale Herleitung eines Algorithmus

Ziel ist es nun, Algorithmen zu beschreiben und zu analysieren, welche bei gegebener Markow-Kette  $M$ , gegebenem Startzustand  $s$  und gegebener Zielzustandsmenge  $A$  die Varianz bzw. Kovarianz berechnen. Wir beschränken uns dabei auf den Fall, dass  $A$  von jedem Zustand in  $M$  aus erreichbar ist.

**Definition 3.1** Sei  $Q$  eine Menge,  $n \in \mathbb{N}, n > 2$  und  $p \in Q^n$ , in folgenden Betrachtungen als Pfad aufgefasst. Dann bezeichnen wir mit  $p_{\leftarrow 1}$  den Teilpfad von  $p = (p_0, \dots, p_{n-1})$  ohne den ersten Knoten  $p_0$ :

$$p_{\leftarrow 1} := (p_1, p_2, \dots, p_{n-1}) \quad (16)$$

#### 3.1 Berechnung von Erwartungswerten

Seien  $M = (Q, P, I)$  eine Markow-Kette,  $s \in Q$ ,  $\emptyset \neq A \subseteq Q$  und sei von jedem Zustand  $q \in Q$  ein Zustand in  $A$  erreichbar. Sei  $R$  eine Gewichtsfunktion auf  $M$ . Wir betrachten im diskreten Wahrscheinlichkeitsraum  $(\text{Paths}_{s \rightarrow A}(M), P)$  den Erwartungswert  $\mathcal{E}_{(\text{Paths}_{s \rightarrow A}(M), P)}(R)$ , im Folgenden kurz  $\mathcal{E}_s(R)$ :

$$\mathcal{E}_s(R) = \sum_{p \in \text{Paths}_{s \rightarrow A}(M)} P(p) \cdot R(p) \quad (17)$$

Falls  $s \in A$ , gilt  $|\text{Paths}_{s \rightarrow A}(M)| = 1$  mit dem einzigen enthaltenen Pfad  $p = (s)$ . Dann sind nach Definition  $P(p) = 1$  und  $R(p) = 0$ . Damit erhalten wir:

$$\mathcal{E}_s(R) = 0 \quad (\text{falls } s \in A) \quad (18)$$

Falls  $s \notin A$ , besteht jeder Pfad  $p \in \text{Paths}_{s \rightarrow A}(M)$  aus mehr als einem Knoten,

und wir erhalten:

$$\mathcal{E}_s(R) = \sum_{p \in \text{Paths}_{s \rightarrow A}(M)} P(s, p_1) \cdot P(p_{\leftarrow 1}) \cdot (R(s, p_1) + R(p_{\leftarrow 1})) \quad (19)$$

$$= \sum_{t \in Q} P(s, t) \cdot \sum_{p' \in \text{Paths}_{t \rightarrow A}(M)} P(p') \cdot (R(s, t) + R(p')) \quad (20)$$

$$= \sum_{t \in Q} P(s, t) \cdot \left( R(s, t) + \sum_{p' \in \text{Paths}_{t \rightarrow A}(M)} P(p') \cdot R(p') \right) \quad (21)$$

$$= \sum_{t \in Q} P(s, t) \cdot (R(s, t) + \mathcal{E}_t(R)) \quad (22)$$

Die Gleichungen 18 und 22 geben uns ein System von  $|Q|$  linearen Gleichungen in  $|Q|$  Variablen:

$$\begin{aligned} \mu_s &= 0 \quad (\text{falls } s \in A) \\ \mu_s &= \sum_{t \in Q} P(s, t) \cdot (R(s, t) + \mu_t) \quad (\text{falls } s \notin A) \end{aligned} \quad (23)$$

Mit Definition 2.19 ist dieses Gleichungssystem äquivalent zu

$$\mu_s = \sum_{t \in Q} P_{\rightarrow A}(s, t) \cdot (R(s, t) + \mu_t), \quad (24)$$

was sich in Matrixschreibweise mit  $\mu = (\mu_s)_{s \in Q}$  ausdrücken lässt als:

$$\mu = \left( \sum_{t \in Q} P_{\rightarrow A}(s, t) \cdot R(s, t) \right)_{s \in Q} + P_{\rightarrow A} \cdot \mu \quad (25)$$

**Satz 3.2 (Erwartungswerte in Markow-Ketten)** Seien  $M = (Q, P, I)$  eine Markow-Kette und  $\emptyset \neq A \subseteq Q$  und sei von jedem Zustand  $q \in Q$  ein Zustand in  $A$  erreichbar. Dann ist der Vektor  $\mu = (\mu_s)_{s \in Q}$  der Erwartungswerte akkumulierter Kantengewichte einer Gewichtsfunktion  $R$  auf Pfaden von  $s$  in die Menge  $A$  die eindeutige Lösung des Gleichungssystems

$$(P_{\rightarrow A} - \mathbb{1})\mu = - \left( \sum_{t \in Q} P_{\rightarrow A}(s, t) \cdot R(s, t) \right)_{s \in Q}. \quad (26)$$

*Beweis.* Wir stellen fest, der Vektor  $(\mathcal{E}_s(R))_{s \in Q}$  ist die einzige Lösung des Gleichungssystems (26): Zum einen ist gemäß unserer Herleitung der Vektor  $(\mathcal{E}_s(R))_{s \in Q}$  eine Lösung dieses Gleichungssystems. Zum anderen ist  $P_{\rightarrow A} - \mathbb{1}$  nach Satz 2.20 invertierbar. Somit ist die Lösung eindeutig.  $\square$

Mit dem Gleichungssystem (26) und Standardalgorithmen zum Lösen linearer Gleichungssysteme erhalten wir unmittelbar einen Algorithmus zur Berechnung der Erwartungswerte  $\mathcal{E}_q(R)$  für  $q \in Q$ .

**Beispiel 3.3** Wir wollen nun Beispiel 2.18 fortsetzen und den Erwartungswert mittels linearem Gleichungssystem lösen. Seien also  $M$  und  $R$  wie bisher definiert. Dann ergibt sich die Matrix  $P_{\rightarrow\{5,6\}} - \mathbb{1}$  als

$$P_{\rightarrow\{5,6\}} - \mathbb{1} = \begin{pmatrix} -1 & 0,5 & 0,5 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 1 \\ 0 & 0,25 & -1 & 0,5 & 0,25 & 0 \\ 0 & 0 & 0 & -0,2 & 0,2 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \end{pmatrix}.$$

Als Inverse von  $P_{\rightarrow\{5,6\}} - \mathbb{1}$  erhalten wir

$$(P_{\rightarrow\{5,6\}} - \mathbb{1})^{-1} = \begin{pmatrix} -1 & -\frac{5}{8} & -\frac{1}{2} & -\frac{5}{4} & -\frac{3}{8} & -\frac{5}{8} \\ 0 & -1 & 0 & 0 & 0 & -1 \\ 0 & -\frac{1}{4} & -1 & -\frac{5}{2} & -\frac{3}{4} & -\frac{1}{4} \\ 0 & 0 & 0 & -5 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \end{pmatrix}.$$

Außerdem berechnen wir wie in Satz 3.2 beschrieben einen Vektor:

$$\begin{aligned} b &:= - \left( \sum_{t \in Q} P_{\rightarrow\{5,6\}}(s, t) \cdot R(s, t) \right)_{s \in Q} \\ &= \left( -\frac{9}{2} \quad -2 \quad -3 \quad -\frac{11}{5} \quad 0 \quad 0 \right)^T \end{aligned}$$

Wir erhalten in der Folge für das Produkt  $\mu := (P_{\rightarrow\{5,6\}} - \mathbb{1})^{-1}b$  den Vektor

$$\mu = (10 \quad 2 \quad 9 \quad 11 \quad 0 \quad 0)^T.$$

Auf diese Weise haben wir den Erwartungswert aufsummierter Kantengewichte entlang von Pfaden von Knoten 1 bis zum Erreichen eines Knotens in  $\{5, 6\}$ , nämlich 10 berechnet, kennen jedoch gleichzeitig schon alle Erwartungswerte für den Start in einem anderen Knoten.

### 3.2 Berechnung von Varianzen

Die Betrachtung zu Kovarianzen sind weitestgehend analog zu den Betrachtungen zu Varianzen, welche wir hier darstellen wollen. Man kann Satz 3.4, welchen wir in diesem Abschnitt zeigen wollen, auch als Korollar des Satzes 3.6 auffassen, welcher im nachfolgenden Abschnitt vorgestellt wird. Man kann also bei Bedarf diesen Abschnitt überspringen. Dennoch wollten wir der Anschaulichkeit wegen den etwas einfacheren Fall der Varianz voranstellen.

Seien  $M = (Q, P, I)$  eine Markow-Kette,  $s \in Q$ ,  $\emptyset \neq A \subseteq Q$  und sei von jedem Zustand  $q \in Q$  ein Zustand in  $A$  erreichbar. Sei  $R$  eine Gewichtsfunktion auf  $M$ . Wir betrachten in  $(\text{Paths}_{s \rightarrow A}(M), P)$  die Varianz  $\mathcal{V}ar_{(\text{Paths}_{s \rightarrow A}(M), P)}(R)$ :

$$\mathcal{V}ar_{(\text{Paths}_{s \rightarrow A}(M), P)}(R) = \mathcal{E}_{(\text{Paths}_{s \rightarrow A}(M), P)} \left( (R - \mathcal{E}_{(\text{Paths}_{s \rightarrow A}(M), P)}(R))^2 \right) \quad (27)$$

Erneut nutzen wir die Kurzschreibweise  $\mathcal{V}ar_s(R) := \mathcal{V}ar_{(\text{Paths}_{s \rightarrow A}(M), P)}(R)$  und erhalten:

$$\mathcal{V}ar_s(R) = \mathcal{E}_s \left( (R - \mathcal{E}_s(R))^2 \right) \quad (28)$$

Falls  $s \in A$ , gilt  $|\text{Paths}_{s \rightarrow A}(M)| = 1$  mit dem einzigen enthaltenen Pfad  $p = (s)$ . Dann sind nach Definition  $P(p) = 1$  und  $R(p) = 0$  und der Erwartungswert beträgt  $\mathcal{E}_s(R) = 0$ , wie wir bereits im vorangegangenen Abschnitt gesehen haben. Damit erhalten wir:

$$\text{Var}_s(R) = 0 \quad (\text{falls } s \in A) \quad (29)$$

Falls  $s \notin A$ , besteht jeder Pfad  $p \in \text{Paths}_{s \rightarrow A}(M)$  aus mehr als einem Knoten, und wir erhalten:

$$\text{Var}_s(R) = \sum_{p \in \text{Paths}_{s \rightarrow A}(M)} P(p) \cdot (R(p) - \mathcal{E}_s(R))^2 \quad (30)$$

$$= \sum_{t \in Q} P(s, t) \cdot \sum_{p' \in \text{Paths}_{t \rightarrow A}(M)} P(p') \cdot (R(s, t) + R(p') - \mathcal{E}_s(R))^2 \quad (31)$$

$$= \sum_{t \in Q} P(s, t) \cdot \mathcal{E}_t \left( (R + R(s, t) - \mathcal{E}_s(R))^2 \right) \quad (32)$$

$$= \sum_{t \in Q} P(s, t) \cdot \left( \text{Var}_t(R) + \left( \mathcal{E}_t(R + R(s, t) - \mathcal{E}_s(R)) \right)^2 \right) \quad (33)$$

$$= \sum_{t \in Q} P(s, t) \cdot \left( \text{Var}_t(R) + (\mathcal{E}_t(R) + R(s, t) - \mathcal{E}_s(R))^2 \right) \quad (34)$$

Gleichung (33) erhalten wir durch Anwendung von Korollar 2.10 und Gleichung (34) durch die Linearität des Erwartungswertes nach Lemma 2.6. Die Gleichungen (29) und (34) geben uns analog zum Abschnitt über die Erwartungswerte ein System von  $|Q|$  linearen Gleichungen in  $|Q|$  Variablen:

$$\begin{aligned} \nu_s &= 0 \quad (\text{falls } s \in A) \\ \nu_s &= \sum_{t \in Q} P(s, t) \cdot \left( \nu_t + (\mathcal{E}_t(R) + R(s, t) - \mathcal{E}_s(R))^2 \right) \quad (\text{falls } s \notin A) \end{aligned} \quad (35)$$

**Satz 3.4 (Varianzen in Markow-Ketten)** *Sei  $M = (Q, P, I)$  eine Markow-Kette. Sei  $\emptyset \neq A \subseteq Q$  und sei von jedem Zustand  $q \in Q$  ein Zustand in  $A$  erreichbar. Dann ist der Vektor  $\nu = (\nu_s)_{s \in Q} = (\text{Var}_s(R))_{s \in Q}$  der Varianzen akkumulierter Kantengewichte einer Gewichtsfunktion  $R$  auf Pfaden von  $s$  in die Menge  $A$  die eindeutige Lösung des Gleichungssystems*

$$(P_{\rightarrow A} - \mathbb{1})\nu = - \left( \sum_{t \in Q} P_{\rightarrow A}(s, t) \cdot S(s, t) \right)_{s \in Q}, \quad (36)$$

wobei  $S$  die Gewichtsfunktion definiert durch  $S : Q^2 \rightarrow \mathbb{R}_+ : (s, t) \mapsto (\mathcal{E}_t(R) + R(s, t) - \mathcal{E}_s(R))^2$  ist.

*Beweis.* Durch Anwendung von Definition 2.19 auf Gleichung (35) erhalten wir direkt dieses Gleichungssystem. Daher ist der Vektor der Varianzen tatsächlich eine Lösung. Nach Satz 2.20 ist die Lösung eindeutig.  $\square$

Vergleicht man die Gleichungssysteme (26) und (36), dann bekommen wir:

$$\forall q \in Q : \mathcal{V}ar_q(R) = \mathcal{E}_q(S) \quad (37)$$

Damit haben wir die Berechnung der Varianzen gleichzeitig auf die Berechnung von Erwartungswerten zurückgeführt. Um die Varianzen  $(\mathcal{V}ar_q(R))_{q \in Q}$  in einer Markow-Kette zu berechnen, genügt es die Erwartungswerte  $(\mathcal{E}_q(R))_{q \in Q}$  im ersten Schritt und  $(\mathcal{E}_q(S))_{q \in Q}$  im zweiten Schritt zu berechnen. Berechnen wir zuerst die inverse Matrix  $(P_{\rightarrow A} - \mathbb{1})^{-1}$ , dann müssen wir zum Lösen beider Gleichungssysteme jeweils nur eine Matrixmultiplikation durchführen.

**Beispiel 3.5** Wir wollen erneut an unsere bisherige konkrete Markow-Kette anknüpfen und setzen die Beispiele 2.18 und 3.3 fort. Zunächst berechnen wir die konkrete Gewichtsfunktion  $S$ , definiert durch

$$S : Q^2 \rightarrow \mathbb{R}_+ : (s, t) \mapsto (\mathcal{E}_t(R) + R(s, t) - \mathcal{E}_s(R))^2.$$

Diese lässt sich als Matrix notieren:

$$S = \begin{pmatrix} 0 & 16 & 16 & 1 & 100 & 100 \\ 64 & 0 & 49 & 81 & 4 & 0 \\ 1 & 4 & 0 & 16 & 36 & 81 \\ 1 & 81 & 4 & 4 & 64 & 121 \\ 100 & 4 & 81 & 121 & 0 & 9 \\ 100 & 4 & 81 & 121 & 49 & 0 \end{pmatrix}$$

Tatsächlich sind für die weitere Berechnung bei Weitem nicht alle Einträge der Matrix relevant, sondern nur die für solche Paare  $(s, t)$ , die Kanten in  $P_{\rightarrow \{5,6\}}$  darstellen, d.h.  $(s, t) \in \text{supp}(P_{\rightarrow \{5,6\}})$ . Es reicht also eine Matrix  $\tilde{S}$  mit

$$\tilde{S} = \begin{pmatrix} 0 & 16 & 16 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & 16 & 36 & 0 \\ 0 & 0 & 0 & 4 & 64 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

Dann berechnen wir analog zu Beispiel 3.3 nach Satz 3.4 einen Vektor:

$$\begin{aligned} c &:= - \left( \sum_{t \in Q} P_{\rightarrow \{5,6\}}(s, t) \cdot S(s, t) \right)_{s \in Q} \\ &= - \left( \sum_{t \in Q} P_{\rightarrow \{5,6\}}(s, t) \cdot \tilde{S}(s, t) \right)_{s \in Q} \\ &= (-16 \quad 0 \quad -18 \quad -16 \quad 0 \quad 0)^\top \end{aligned}$$

Wir erhalten in der Folge für das Produkt  $\nu := (P_{\rightarrow \{5,6\}} - \mathbb{1})^{-1} c$  den Vektor

$$\nu = (45 \quad 0 \quad 58 \quad 80 \quad 0 \quad 0)^\top.$$

Im ersten Eintrag des Vektors sehen wir dieselbe Varianz, welche wir bereits in Beispiel 2.18 ermittelt haben. Die Varianz für den Startknoten 2 ist 0 in

Übereinstimmung mit der Eigenschaft, dass es von dort aus nur einen eindeutigen Pfad gibt und somit die Abweichung vom Erwartungswert in jedem Fall 0 beträgt. Die Varianzen für Knoten 3 und 4 sind entsprechend  $\mathcal{V}ar_3(R) = 58$  bzw.  $\mathcal{V}ar_4(R) = 80$ .

### 3.3 Berechnung von Kovarianzen

Seien wieder  $M = (Q, P, I)$  eine Markow-Kette,  $s \in Q$ ,  $\emptyset \neq A \subseteq Q$  und sei von jedem Zustand  $q \in Q$  ein Zustand in  $A$  erreichbar. Seien nun  $X, Y$  Gewichtsfunktionen auf  $M$ . Wir betrachten im diskreten Wahrscheinlichkeitsraum  $(\text{Paths}_{s \rightarrow A}(M), P)$  die Kovarianz  $\text{Cov}_s(X, Y) := \text{Cov}_{(\text{Paths}_{s \rightarrow A}(M), P)}(X, Y)$ :

$$\text{Cov}_s(X, Y) = \mathcal{E}_s((X - \mathcal{E}_s(X))(Y - \mathcal{E}_s(Y))) \quad (38)$$

Falls  $s \in A$ , gilt  $|\text{Paths}_{s \rightarrow A}(M)| = 1$  mit dem einzigen enthaltenen Pfad  $p = (s)$ . Dann sind nach Definition  $P(p) = 1$  und  $R(p) = 0$  und die Erwartungswerte betragen  $\mathcal{E}_s(X) = \mathcal{E}_s(Y) = 0$ . Damit erhalten wir:

$$\text{Cov}_s(X, Y) = 0 \quad (\text{falls } s \in A) \quad (39)$$

Falls  $s \notin A$ , besteht jeder Pfad  $p \in \text{Paths}_{s \rightarrow A}(M)$  aus mehr als einem Knoten und wir erhalten:

$$\text{Cov}_s(X, Y) = \sum_{p \in \text{Paths}_{s \rightarrow A}(M)} P(p) \cdot ((X(p) - \mathcal{E}_s(X))(Y(p) - \mathcal{E}_s(Y))) \quad (40)$$

$$= \sum_{t \in Q} P(s, t) \sum_{p' \in \text{Paths}_{t \rightarrow A}(M)} P(p') \cdot \left( (X(s, t) + X(p') - \mathcal{E}_s(X)) \cdot (Y(s, t) + Y(p') - \mathcal{E}_s(Y)) \right) \quad (41)$$

$$= \sum_{t \in Q} P(s, t) \cdot \mathcal{E}_t \left( (X + X(s, t) - \mathcal{E}_s(X)) \cdot (Y + Y(s, t) - \mathcal{E}_s(Y)) \right) \quad (42)$$

$$= \sum_{t \in Q} P(s, t) \cdot \left( \begin{aligned} &\mathcal{E}_t(XY) + \mathcal{E}_t(X)(Y(s, t) - \mathcal{E}_s(Y)) + \\ &\quad + \mathcal{E}_t(Y)(X(s, t) - \mathcal{E}_s(X)) + \\ &\quad + (X(s, t) - \mathcal{E}_s(X))(Y(s, t) - \mathcal{E}_s(Y)) \end{aligned} \right) \quad (43)$$

$$= \sum_{t \in Q} P(s, t) \cdot \left( \begin{aligned} &\mathcal{E}_t(XY) - \mathcal{E}_t(X)\mathcal{E}_t(Y) + \\ &\quad + \mathcal{E}_t(X)\mathcal{E}_t(Y) + \mathcal{E}_t(X)(Y(s, t) - \mathcal{E}_s(Y)) + \\ &\quad + \mathcal{E}_t(Y)(X(s, t) - \mathcal{E}_s(X)) + \\ &\quad + (X(s, t) - \mathcal{E}_s(X))(Y(s, t) - \mathcal{E}_s(Y)) \end{aligned} \right) \quad (44)$$

$$= \sum_{t \in Q} P(s, t) \cdot \left( \text{Cov}_t(X, Y) + \left( \mathcal{E}_t(X) + X(s, t) - \mathcal{E}_s(X) \right) \cdot \left( \mathcal{E}_t(Y) + Y(s, t) - \mathcal{E}_s(Y) \right) \right) \quad (45)$$

Gleichung (41) erhalten wir durch Partitionierung der Pfade nach der ersten Transition. Mit Definition 2.3 erhalten wir Gleichung (42). Gleichung (43) erhalten wir durch partielles Ausmultiplizieren und anschließendes Anwenden der Linearität (Lemma 2.6). Um Gleichung (44) zu erhalten, addieren wir an geeigneter Stelle  $-\mathcal{E}_t(X)\mathcal{E}_t(Y) + \mathcal{E}_t(X)\mathcal{E}_t(Y) = 0$ . Danach können wir die ersten

zwei Summanden der großen Klammer nach Lemma 2.9 als Kovarianz auffassen sowie den Rest durch geschicktes Ausklammern kompakt darstellen und erhalten Gleichung (45).

Sei die Gewichtsfunktion  $S$  auf  $M$  definiert als

$$S : Q^2 \rightarrow \mathbb{R} : (s, t) \mapsto (\mathcal{E}_t(X) + X(s, t) - \mathcal{E}_s(X))(\mathcal{E}_t(Y) + Y(s, t) - \mathcal{E}_s(Y)). \quad (46)$$

Die Gleichungen (39) und (45) geben uns ähnlich den Abschnitten über die Erwartungswerte bzw. Kovarianzen ein System von  $|Q|$  linearen Gleichungen in  $|Q|$  Variablen:

$$\begin{aligned} c_s &= 0 \quad (\text{falls } s \in A) \\ c_s &= \sum_{t \in Q} P(s, t) \cdot (c_t + S(s, t)) \quad (\text{falls } s \notin A) \end{aligned} \quad (47)$$

**Satz 3.6 (Kovarianzen in Markow-Ketten)** *Seien  $M = (Q, P, I)$  eine Markow-Kette und  $\emptyset \neq A \subseteq Q$ . Sei von jedem Zustand  $q \in Q$  ein Zustand in  $A$  erreichbar. Dann ist der Vektor  $c = (c_s)_{s \in Q} = (\text{Cov}_s(X, Y))_{s \in Q}$  der Kovarianzen akkumulierter Kantengewichte bezüglich Gewichtsfunktionen  $X, Y$  auf Pfaden von  $s$  in die Menge  $A$  die eindeutige Lösung des Gleichungssystems*

$$(P_{\rightarrow A} - \mathbb{1})c = - \left( \sum_{t \in Q} P_{\rightarrow A}(s, t) \cdot S(s, t) \right)_{s \in Q}, \quad (48)$$

wobei  $S$  die in Gleichung (46) definierte aus  $X$  und  $Y$  abgeleitete Kantengewichtsfunktion bezeichnet.

*Beweis.* Analog zu vorherigen Abschnitten erhalten wir durch Anwendung von Definition 2.19 auf das Gleichungssystem (47), dass der Vektor der Kovarianzen in der Tat Lösung des dargestellten Gleichungssystems ist. Wieder besitzt das Gleichungssystem eine eindeutige Lösung nach Satz 2.20.  $\square$

Vergleicht man die Gleichungssysteme (26) und (48), dann bekommen wir:

$$\forall q \in Q : \text{Cov}_q(X, Y) = \mathcal{E}_q(S) \quad (49)$$

Das entspricht dem Zurückführen der Berechnung von Kovarianzen auf die Berechnung von Erwartungswerten. Um die Kovarianzen  $(\text{Cov}_q(X, Y))_{q \in Q}$  in einer Markow-Kette zu berechnen, genügt es die Erwartungswerte  $(\mathcal{E}_q(X))_{q \in Q}$  sowie  $(\mathcal{E}_q(Y))_{q \in Q}$  im ersten Schritt und  $(\mathcal{E}_q(S))_{q \in Q}$  im zweiten Schritt zu berechnen. Es ist wie bei den Varianzen möglich, zuerst die inverse Matrix  $(P_{\rightarrow A} - \mathbb{1})^{-1}$  zu berechnen, um danach nur noch Matrixmultiplikationen ausführen zu müssen.

### 3.4 Die initiale Verteilung

Wir haben bisher nur über die Erwartungswerte und (Ko-) Varianzen für den Start in einem speziellen Zustand gesprochen. Wenn es darum geht, diese Kenngrößen einer Markow-Kette  $M = (Q, P, I)$  insgesamt zu berechnen, muss aber auch die initiale Zufallsverteilung  $I : Q \rightarrow [0, 1]$  in Betracht gezogen werden. Der Erwartungswert der Markow-Kette ist als die gewichtete Summe  $\mu = \sum_{q \in Q} I(q) \cdot \mu_q$  definiert. Um nicht Erwartungswert und (Ko-) Varianz der gesamten Markow-Kette auf diese Weise gesondert betrachten zu müssen, können wir uns dem Trick bedienen, mit  $Q' := Q \cup \{q_{\text{start}}\}$  einen neuen Zustand  $q_{\text{start}} \notin Q$  hinzuzufügen. Mit  $I'$  und  $P'$  definiert als



$$I'(q) := \begin{cases} 1 & q = q_{start} \\ 0 & q \neq q_{start} \end{cases} \quad P'(s, t) := \begin{cases} I(t) & s = q_{start} \wedge t \in Q \\ 0 & t = q_{start} \\ P(s, t) & s, t \in Q \end{cases}$$

erhalten wir eine neue Markow-Kette  $M' = (Q', P', I')$ , wobei Erwartungswert und (Ko-) Varianz von  $M$  und  $M'$  mit den entsprechenden Werten von  $q_{start}$  übereinstimmen.

Wir wollen uns deshalb mit der zustandsweisen Betrachtung der Größen begnügen. Ferner können wir die genannten Größen neben alternativen Möglichkeiten mithilfe genau dieser Betrachtung definieren. Seien dazu  $R, S$  Gewichtsfunktionen und  $A$  eine Zielzustandsmenge. Wir erweitern  $R$  entsprechend so, dass  $R(s, t) = 0$ , wenn  $q_{start} \in \{s, t\}$  und analog für  $S$ . Wir definieren Erwartungswert, Varianz und Kovarianz für  $M$  als:

$$\begin{aligned} \mathcal{E}(R) &:= \mathcal{E}_{(\text{Paths}_{q_{start} \rightarrow A}(M'), P')}(R) \\ \mathcal{V}ar(R) &:= \mathcal{V}ar_{(\text{Paths}_{q_{start} \rightarrow A}(M'), P')}(R) \\ \mathcal{C}ov(R, S) &:= \mathcal{C}ov_{(\text{Paths}_{q_{start} \rightarrow A}(M'), P')}(R, S) \end{aligned}$$

### 3.5 Komplexität

Zum Berechnen von Erwartungswerten, Varianzen bzw. Kovarianzen unter Zuhilfenahme der gezeigten linearen Gleichungssysteme sind mehrere Schritte notwendig:

- Zunächst berechnen wir die Matrix  $M := (P_{\rightarrow A} - \mathbb{1})$ .
- Danach berechnen wir  $b := - \left( \sum_{t \in Q} P_{\rightarrow A}(s, t) \cdot R(s, t) \right)_{s \in Q}$ .
- Anschließend lösen wir das lineare Gleichungssystem  $Mx = b$ , wobei gilt  $x = M^{-1}b$ .

Für das Berechnen von Erwartungswerten müssen wir nichts weiter tun, denn es gilt  $\forall s \in Q : x(s) = \mathcal{E}_s(R)$ . Für die Berechnung von Varianzen und Kovarianzen müssen wir die Schritte lediglich mehrfach ausführen, um zunächst die Erwartungswerte zu berechnen und danach aus diesen einen neuen Reward (Gewichtsfunktion) zu definieren wie beispielsweise die Funktion  $S$  in Satz 3.4.

Das Berechnen des neuen Rewards ist in  $\mathcal{O}(\#edges)$  möglich, wobei  $\#edges$  die Zahl der Kanten in der Markow-Kette bezeichnet. Die Ermittlung von  $M$  ist in  $\mathcal{O}(|Q|^2)$  möglich. Geht man von dünn besetzten Matrizen aus, so ist bei entsprechender Speicherverwaltung die Berechnung sogar in  $\mathcal{O}(\#edges)$  möglich, da nur über echte Kanten iteriert werden muss, nicht aber über entartete Kanten  $(s, t)$  im Sinne von  $P(s, t) = 0$ . Den Vektor  $b$  können wir in  $\mathcal{O}(|Q|^2)$  Schritten berechnen, bzw. analog in  $\mathcal{O}(\#edges)$  bei einer dünn besetzten Matrix und entsprechender Speicherung.

Das Lösen linearer Gleichungssysteme ist ein Standardproblem und wurde dementsprechend ausführlich untersucht. Ohne ins Detail zu gehen, können wir festhalten, dass sich das lineare Gleichungssystem  $Mx = b$  mit  $M \in \mathbb{R}^{|Q| \times |Q|}$  in  $\mathcal{O}(|Q|^3)$  lösen lässt unter der Annahme, dass jede einzelne Additions- und

Multiplikationsoperation in einem Zeitschritt erfolgt. Ein Beispiel für einen Algorithmus ist das Gaußsche Eliminationsverfahren. Will man allerdings ein solches Gleichungssystem unter Verwendung von rationalen Zahlen explizit und ohne Verlust von Genauigkeit lösen, dann benötigen einzelne Additions- bzw. Multiplikationsschritte mit herkömmlicher Rechentechnik bei Verwendung einer Darstellung mittels BigInteger im Allgemeinen mehr als nur einen Taktzyklus des Prozessors. Ein Algorithmus [3] arbeitet unter einem Zeitaufwand in  $\mathcal{O}(n^3(\log n)^2)$  unter Nutzung von p-adischen Körpererweiterungen von  $\mathbb{Q}$ .

Auch kann das Lösen nicht in weniger als  $\mathcal{O}(\#edges)$  Schritten möglich sein, da es mindestens so viele Schritte benötigt, die Matrix  $M$  überhaupt einzulesen. Dies gibt uns eine feste untere Schranke für alle Algorithmen.

Wir können zusammenfassen, dass der Zeitaufwand für die Berechnung von Erwartungswerten bzw. (Ko-) Varianzen im Wesentlichen vom Algorithmus zum Lösen des linearen Gleichungssystems abhängt, da der asymptotische Aufwand zur Berechnung von  $M$  und  $b$  höchstens genauso groß oder geringer ist. Ausgehend von bekannten Verfahren zur expliziten Lösung linearer Gleichungssysteme ist er wesentlich geringer.

## 4 Performancemessung am praktischen Beispiel

Wir wollen die beschriebene Berechnung von Erwartungswerten, Varianzen und Kovarianzen praktisch an zwei Beispielen demonstrieren. Als Beispiele dienen uns Modelle, welche unter anderem in den *PRISM Case Studies* [7] zu finden sind. Zum einen wollen wir *Herman's self-stabilising algorithm* [4], kurz *herman*, zum anderen das *Synchronous Leader Election Protocol* von Itai und Rodeh [5], kurz *leader\_sync*, betrachten. Ersteres Beispiel wollen wir zur Veranschaulichung hier einmal ausführlich beschreiben, für *leader\_sync* zumindest die Intuition angeben.

### 4.1 Herman's selbst-stabilisierender Algorithmus

Das zu lösende Problem besteht allgemein formuliert darin, dass ein Netzwerk von  $n$  identischen Prozessen, welche allesamt mit einer möglicherweise instabilen Konfiguration starten, mit Wahrscheinlichkeit 1 nach endlich vielen Schritten des Algorithmus' einen stabilen Zustand erreicht.

**Beispiel 4.1 (Stabilisierung im unidirektionalen Ring)** Wir nutzen im Folgenden zur Vereinfachung die Abkürzung  $\underline{n} = \{0, \dots, n-1\}$  und insbesondere  $\underline{2}$  für die Menge der booleschen Werte, wobei wir *true* mit 1 und *false* mit 0 identifizieren. Sei  $n \in 2\mathbb{N} + 1$  eine ungerade natürliche Zahl. Dann betrachten wir einen Ring von  $n$  Prozessen  $P_0, P_1, \dots, P_{n-1}$ , in welchem unidirektionale Kommunikation stattfindet. Ein Zustand  $z$  ist eine Funktion, die jedem Prozess einen booleschen Wert zuordnet, d.h.  $z : \underline{n} \rightarrow 2$ . Gesucht ist ein Algorithmus, sodass nach endlich vielen Schritten fast sicher nur noch Zustände folgen, in welchen genau ein Prozess als privilegierter Prozess ausgewählt ist, oder anders ausgedrückt  $\exists i \in \underline{n} : z(i) \wedge \forall j \in \underline{n} : i \neq j \Rightarrow \neg z(j)$ .

Herman hat dazu die folgende Lösung präsentiert:

**Algorithmus 4.2** Sei  $y$  eine Funktion vom gleichen Typ wie  $z$ , also  $y : \underline{n} \rightarrow 2$  und sei  $i \in \underline{n}$  eine Prozess-ID. Wir fassen  $y(i)$  als den internen Zustand des

Prozesses  $i$  auf sowie  $y(i - 1 \bmod n)$  als den Zustand von dessen Vorgänger, welcher dem Prozess  $i$  über die Ringinfrastruktur bekannt wird. Jeder einzelne Prozess führt nun in einem Schritt Folgendes aus:

- Falls  $y(i - 1 \bmod n) \neq y(i)$ , setze für den Folgezustand  $y'(i) := \neg y(i)$ .
- Falls  $y(i - 1 \bmod n) = y(i)$ , gehe mit Wahrscheinlichkeit 0,5 in den Folgezustand mit  $y'(i) := 0$  und sonst in den Folgezustand mit  $y'(i) := 1$ .

Sei weiterhin  $f$  die Funktion

$$f : \underline{2}^n \rightarrow \underline{2}^n : f(y)(i) := \begin{cases} 1 & y(i) = y(i - 1 \bmod n) \\ 0 & \text{sonst} \end{cases}, \quad (50)$$

wobei  $\underline{2}^n$  die Menge aller Funktionen  $y : \underline{n} \rightarrow \underline{2}$  bezeichnet. Mit  $z := f(y)$  erhalten wir einen prozessbezogenen booleschen Wert, der die gewünschte Bedingung erfüllt, also der bei Grenzwertbetrachtung mit Wahrscheinlichkeit 1 nach endlichen Schritten nur für genau einen Prozess 1 ergibt. Wohlgemerkt kann jeder Prozess  $P_i$  seinen Wert  $z(i)$  berechnen, da er dafür nur bereits bekannte Stellen von  $y$  kennen muss. Als Intuition kann man formulieren,  $f(y)(i)$  ist genau dann 1, wenn Prozess  $i$  seinen Nachfolgezustand echt probabilistisch aussucht, d.h. mindestens zwei Folgezustände existieren.

Dass dieser Algorithmus die geforderte Eigenschaft erfüllt, wollen wir hier nicht zeigen und verweisen auf Hermans Betrachtungen dazu [4]. Jedoch sei angemerkt, dass es sich relativ leicht sehen lässt, wie von jedem Zustand aus ein stabiler Zustand erreicht werden kann, sowie dass die Menge der stabilen Zustände unter der Zustandsübergangsfunktion abgeschossen ist, welche aus Hermans Algorithmus hervorgeht.

Ein ringförmiges Netzwerk, welches sich nach Hermans Algorithmus verhält, lässt sich als Markow-Kette  $M = (Q, P, I)$  auffassen. Die Zustände  $Q := \underline{2}^n$  sind die Konfigurationen der Prozesse, also Abbildungen von  $\{0, \dots, n-1\}$  in die Menge  $\{true, false\}$ . Wir sehen leicht, dass für jeden Zustand  $y \in Q$  alle ausgehenden Kanten mit dergleichen Wahrscheinlichkeit beschriftet sind, nämlich  $p_y := 2^{-|\text{supp } f(y)|}$ . Der Support einer Funktion  $f : A \rightarrow B$ ,  $0 \in B$ , bezeichnet durch  $\text{supp}(f)$ , ist dabei die Menge aller Urbilder, welche nicht auf 0 abgebildet werden, d.h.  $\text{supp}(f) := \{x \in A \mid f(x) \neq 0\}$ . Die Menge der Folgezustände für ein  $y \in Q$ , bezeichnet durch  $\text{Succ}(y)$ , können wir schreiben als

$$\text{Succ}(y) := \{y' \in Q \mid \neg f(y)(i) \Rightarrow y'(i) \neq y(i)\}.$$

Wir erhalten für  $P$ :

$$P(y, y') := \begin{cases} p_y & \text{falls } y' \in \text{Succ}(y) \\ 0 & \text{falls } y' \notin \text{Succ}(y) \end{cases} \quad (51)$$

Für die initiale Verteilung nehmen wir an, dass alle Prozesse mit einem Zufallsbit starten:

$$\forall y \in Q : I(y) := 2^{-n}$$

Als Menge der Zielzustände  $A = \{y \in Q \mid 1 = |\text{supp } f(y)|\}$  betrachten wir jene Zustände mit genau einem privilegierten Prozess, also nur genau einem

Prozess  $P_i$ , dessen Zustandsbit  $y(i)$  im Folgezustand 0 oder 1 sein kann je nach probabilistischer Entscheidung. Für eine geeignete Gewichtsfunktion weisen wir einfach allen Kanten das Gewicht 1 zu.

Unser zweites Beispiel, *leader\_sync*, welches wir für Messungen heranziehen wollen, basiert ebenfalls auf dem Problem, in einem ringförmigen Netzwerk von  $n$  unabhängigen Prozessoren einen Prozessor auszuwählen. Für *leader\_sync* ist ein zusätzlicher Parameter  $K$  festzulegen. Nach dem Protokoll wählt jeder Prozessor zufällig eine Zahl aus  $\{1, \dots, K\}$  als ID. Wurde eine ID von nur einem Prozessor gewählt, so gewinnt der Prozessor, welcher die größte nicht mehrfach gewählte ID hat. Andernfalls wird eine neue Runde gestartet.

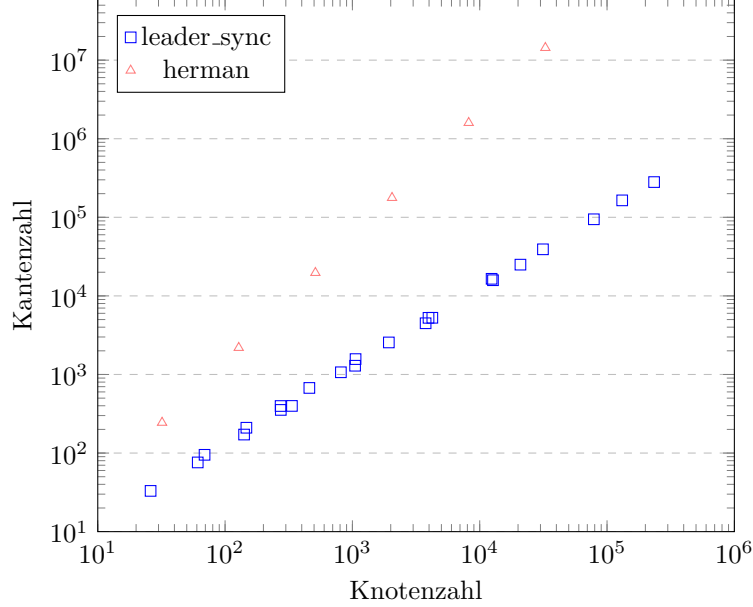
## 4.2 Implementation

Zu dieser schriftlichen Abhandlung wurde eine Implementierung der vorgestellten Berechnungsmethoden angefertigt und ist öffentlich auf *github.com* einsehbar [6]. Wir wollen diese Implementation nutzen, um die Performance der vorgestellten Verfahren an den eben genannten Beispielen zu testen. Zunächst sollen aber hier der wesentliche Aufbau und die wesentlichen Features der Softwarelösung kurz beschrieben werden und auf die verwendeten Bibliotheken eingegangen werden.

Das Tool *MC Analyzer* [6] wurde in der Sprache C++ geschrieben und liefert Algorithmen zur Berechnung von Erwartungswerten, Varianzen und Kovarianzen von akkumulierten Kantengewichten in Markow-Ketten. Dazu muss zunächst eine Markow-Kette geladen werden. Hierfür ist es möglich Dateien, wie *PRISM* [7] sie verwendet, einzulesen oder vom Tool eine Markow-Kette für den Algorithmus von Herman (4.2) erzeugen zu lassen. Weiterhin wird ein eigens mitgebrachtes Dateiformat akzeptiert. Beispiele finden sich auf GitHub. Nachdem eine Markow-Kette und eine Menge an Zielzuständen eingelesen wurden, lassen sich Operationen zur Berechnung von Erwartungswert, Varianz bzw. Kovarianz in einzelnen Schritten ausführen. Dabei schreibt das Tool Messwerte in die Standardausgabe, aus denen unter anderem die Dauer einzelner Berechnungsschritte hervorgeht. Details zur Verwendung sind online beschrieben [6].

Zum Lösen linearer Gleichungssysteme wurde die *header-only C++ library AMGCL* von Demidov [2] verwendet. Diese verwendet algebraische Mehrgitterverfahren zum näherungsweisen Lösen großer linearer Gleichungssysteme mit dünnbesetzter Koeffizientenmatrix  $A_1$ . Ist eine Koeffizientenmatrix  $A_i$  zum direkten Lösen in angemessener Zeit zu groß, wird so lange eine gröbere Matrix  $A_{i+1}$  geringerer Dimension gebildet zusammen mit einem sogenannten Prolongationsoperator  $P_i$  und einem sogenannten Restriktionsoperator  $R_i$  mit  $A_{i+1} = R_i A_i P_i$ , bis wir eine in vernünftiger Zeit direkt lösbare Matrix  $A_n$  erhalten. Nachdem solch eine Hierarchie aufgestellt worden ist, werden zunächst von der feinsten bis zur größten Hierarchieebene jeweils iterativ Näherungslösungen berechnet. Die Fehler werden durch Anwenden der Restriktionsoperation in die jeweils gröbere Hierarchiestufe propagiert. Das lineare Gleichungssystem aus der größten Stufe wird direkt gelöst. Danach werden die Hierarchieebenen in umgekehrter Reihenfolge betrachtet, jeweils die Lösung der gröberen Ebene wird in die feinere propagiert unter Anwendung des Prolongationsoperators. In jeder Ebene werden dabei verbliebene Fehler iterativ minimiert. Nähere Details bietet der Artikel von Demidov [2].

Abbildung 4.1: Kantenzahl in Abhängigkeit der Knotenzahl



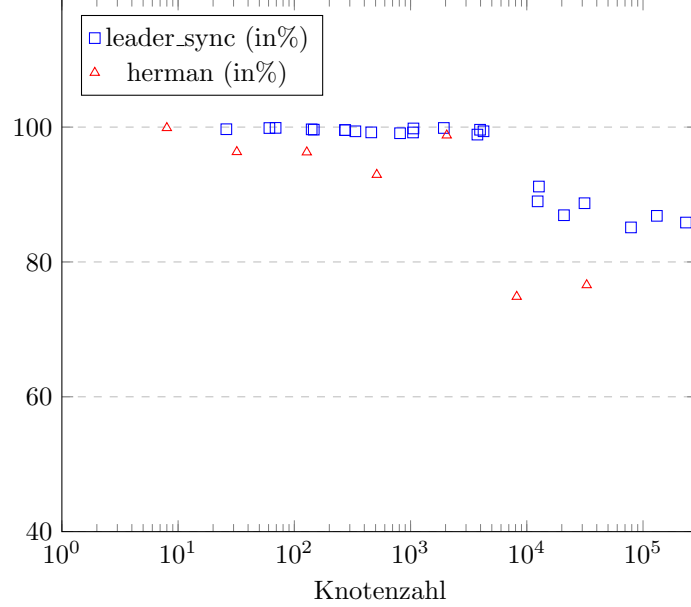
Da in diesem Verfahren einzelne Schritte stets Restfehler verkleinern, tritt im Gegensatz zur klassischen Methode von Gauß-Jordan zum direkten Lösen linearer Gleichungssysteme nicht das typische Problem der numerischen Instabilität in Verbindung mit der floating-point Arithmetik auf. Dieses provozieren wir zum Beispiel, wenn wir zwei positive Zahlen mit sehr geringen Beträgen am Rande des Darstellbaren multiplizieren und in der floating-point Arithmetik als Resultat eine 0 erhalten, da die Genauigkeit nicht mehr ausreicht.

### 4.3 Analyse der Messwerte

Wir sehen im Folgenden Messwerte für die Beispiele *herman* und *leader\_sync*. Gemessen wurde auf einem Windows 10 Pro 1909 mit einem Intel Core i9-9900K mit einer nominalen Taktrate von 3.60GHz. Sowohl *herman* als auch *leader\_sync* geben uns für verschieden große Parameter entsprechend Markow-Ketten in abhängiger Größe. Für *leader\_sync* wurden alle Modelle in Betracht gezogen, welche in den *PRISM Case Studies* [7] zu finden waren. Für *herman* bietet die Software die Möglichkeit, entsprechende Markow-Ketten generieren zu lassen. Es müssen keine Dateien eingelesen werden. Bei *herman* lässt sich beobachten, dass mit  $N = 17$  bereits ein Bedarf an Arbeitsspeicher in der Größenordnung von 60GB besteht. Deswegen konnte nur bis  $N = 15$  gemessen werden. Abbildung 4.1 zeigt die Anzahl der Kanten in der Markow-Kette in Abhängigkeit der Zahl der Knoten. Wir erkennen eine lineare Korrelation zwischen Anzahl der Kanten und Anzahl der Knoten für *leader\_sync*. Für *herman* sehen wir, dass die Anzahl der Kanten in etwa quadratisch wächst bezogen auf die Anzahl der Knoten.

Abbildung 4.2 zeigt zunächst einmal, welchen Anteil das Lösen des linearen Gleichungssystems an der Gesamtzeit der Berechnung hat. Die Gesamtzeit umfasst alle Berechnungsschritte nach vollständigem Einlesen oder Generieren der

Abbildung 4.2: Anteil Lösung des Gleichungssystem an Gesamtaufwand

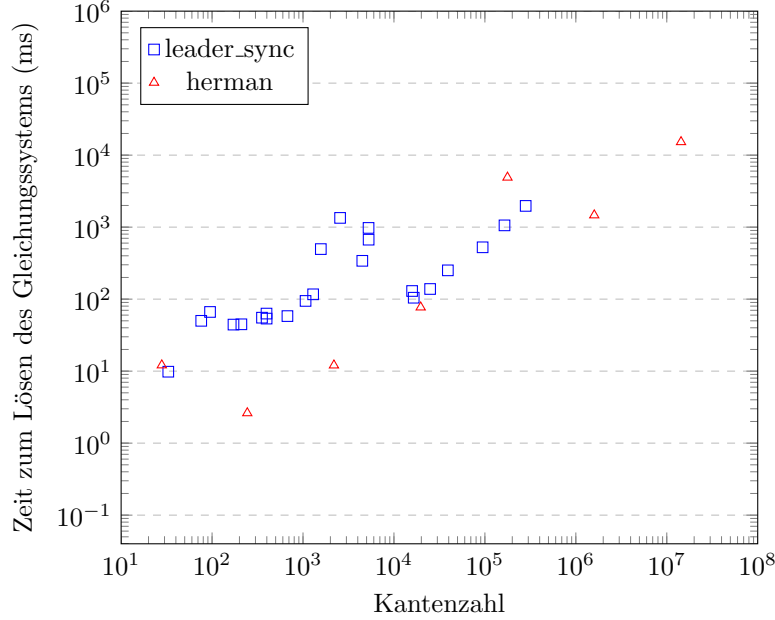


Markow-Kette bis zum Ergebnis. Zur ihr zählen z.B. das Aufstellen der Matrix  $P_{\rightarrow A}$  und das Berechnen der Gewichtsfunktion  $S$  wie in Satz 3.4 verwendet. Nicht inbegriffen ist das Einlesen von Dateien oder Generieren einer Markow-Kette für *herman*. Wir sehen, dass sich für beide Beispiele unsere Erkenntnis aus den theoretischen Betrachtungen bestätigt, nämlich dass die Berechnungszeit im Wesentlichen vom Aufwand zum Lösen des linearen Gleichungssystems abhängt. Wäre dies nicht der Fall, so wäre zu überprüfen, ob die Implementierung hinsichtlich Effizienz im Speicherzugriff größere Fehler beinhaltet.

Nach diesen Vorbetrachtungen wollen wir nun den Zeitaufwand zur Lösung des linearen Gleichungssystems in Abhängigkeit von der Kantenzahl (Abbildung 4.3) betrachten. Sofort erkennen wir, dass beide Messreihen keinen streng monotonen Verlauf aufweisen. Dies ist dadurch zu erklären, dass AMGCL bei gewissen Größen der Markow-Kette beginnt, jeweils weitere Matrizen in der Hierarchie des Mehrgitterverfahrens aufzustellen. Wir sehen für *leader\_sync* drei maximale Intervalle mit (bis auf Ausreißer in der Messung) monoton steigenden Messwerten. Entsprechend wurden keine, genau eine, beziehungsweise genau zwei zusätzliche Matrizen aufgestellt. Diese Zuordnung lässt sich eindeutig beobachten, wenn man die Log-Ausgaben von AMGCL betrachtet. Analog wird für *herman* ab  $N = 13$  eine zweite, größere Matrix aufgestellt. Betrachten wir die Messwerte für größere Kantenzahlen, so lässt sich ablesen, dass der Zeitaufwand in etwa linear von der Kantenzahl abhängt. Schlüsse auf den Verlauf des Graphen für noch größere Eingaben erscheinen auf Basis der Messwerte nicht begründbar und möchten wir hier nicht vornehmen.

Betrachtet haben wir hier bewusst nur die Zeit für den Schritt des Lösens der zwei Gleichungssysteme, nicht jedoch die Gesamtzeit. Man könnte diese Punkte mithilfe von Abbildung 4.2 einzeichnen und würde dann nahezu identische Punkte zu den bereits eingezeichneten erhalten. Insbesondere deswegen wollen wir hier

Abbildung 4.3: Zeitaufwand in Abhängigkeit der Kantenanzahl



darauf verzichten.

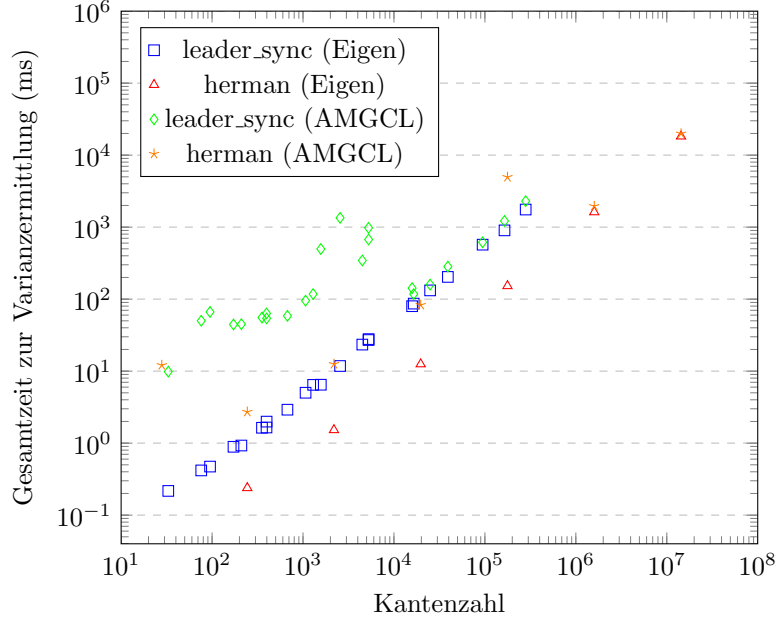
Um noch einen Vergleich anstellen zu können, wurde die verwendete C++ Bibliothek zum Lösen des Gleichungssystems AMGCL einmal durch die Bibliothek Eigen ausgetauscht. Speziell wurde der Algorithmus *BiCGSTAB*, der von Eigen implementiert wird, genutzt, um die Gleichungssysteme mit dünn besetzten Matrizen zu lösen. BiCGSTAB ist dafür bekannt sehr schnell zu laufen, allerdings auch Rundungsfehler bei den Ergebnissen zu produzieren, weswegen er in der Praxis nicht häufig verwendet wird. Dieses Phänomen konnte bei der Entwicklung auch beobachtet werden. Da Eigen selbst Datenstrukturen für die Speicherung von Matrizen mitbringt, welche verwendet werden müssen, um den Algorithmus anzuwenden, und diese entsprechend anders zu initialisieren sind, bietet es sich an, einfach die Gesamtzeit (ohne Einlesen von Dateien) zu messen, welche zur Ermittlung der Varianzen gebraucht wird. Abbildung 4.4 zeigt die Messreihen der Gesamtzeit für *herman* und *leader\_sync*, wobei die Verwendung von Eigen und AMGCL gegenüber gestellt wird. Wie wir sehen, verhält sich der Zeitaufwand bei der Variante Eigen in etwa linear zur Kantenanzahl. Sogar bietet diese Variante kürzere Berechnungszeiten für kleinere Modelle.

Zumindest im Rahmen der Aussagekraft unserer Messdaten lässt sich zusammenfassend erkennen, dass die Zahl der Kanten kennzeichnend für die benötigte Rechenzeit ist und dass letztere sich in etwa linear zur Zahl der Kanten verhält.

#### 4.4 Grenzen der Aussagekraft

Wie bereits kurz zur Erläuterung der Abbildungen erwähnt, steigt der Bedarf an Arbeitsspeicher natürlich recht schnell an, wenn Markow-Ketten eines bestimmten Modells betrachtet werden und die Zahl der Kanten in Abhängigkeit der Parameter des Modells exponentiell wächst. Bei Speicherung der Matrizen

Abbildung 4.4: Zeitaufwand in Abhängigkeit der Kantenanzahl



mittels HashMaps wie im Fall von *MC Analyzer* ist davon auszugehen, dass sich der Speicherbedarf linear zur Zahl der Kanten verhält. Bei Rechenzeiten unter 100s ist hier nicht die Zeit und damit die Taktfrequenz des Prozessors die limitierende Ressource, sondern vielmehr der zur Verfügung stehende Arbeitsspeicher. Ausgaben des Debuggers und Logs von AMGCL zu Folge ist der für das Lösen des linearen Gleichungssystems benötigte RAM nach Aufstellen der Eingabematrix im Vergleich zu dieser Matrix bei den gemessenen Beispielen verschwindend gering. Es genügt nach diesen Beobachtungen im Wesentlichen zu schauen, wie viel Platz zur Darstellung der Markow-Kette im Speicher benötigt wird. Es wäre weiterführend interessant, sich Mittel und Methoden zu überlegen und zu implementieren, den tatsächlichen Speicherbedarf zu messen und in Abhängigkeit von der Zahl der Kanten darzustellen.

Wir haben bei den Messungen gesehen, dass AMGCL bis zu drei Matrizen im Mehrgitterverfahren verwendet. Ob sich die lineare Abhängigkeit der Rechenzeit von der Zahl der Kanten auch bei noch größeren Markow-Ketten und den mit diesen verbundenen Matrizen fortsetzt, lässt sich an dieser Stelle schwer einschätzen. Auch wäre es interessant zu analysieren, wie es sich bei derart größeren Markow-Ketten verhält, bei welchen dann entsprechend mehr Matrizen in der Mehrgitterhierarchie berechnet werden.

Zur Speicherung der Matrizen wurden Hashtabellen genutzt, wie es auch in der Dokumentation von AMGCL vorgeschlagen wird. Die Art und Weise der Speicherung und der damit verbundene Zugriff beeinflusst wesentlich die Laufzeit eines darauf operierenden Algorithmus<sup>7</sup>. Für weiterführenden Untersuchungen wäre es interessant herauszufinden, welche besseren Möglichkeiten es zur Speicherung möglicherweise gibt. Es wäre insbesondere interessant zu prüfen, ob das von der Bibliothek Eigen mitgebrachte Datenformat zur Speicherung von



dünnbesetzten Matrizen auch mit AMGCL verwendet werden kann. Die Speicherung der eingelesenen Markow-Kette selbst müsste dazu angepasst werden, um nicht den erhöhten Zeitaufwand bei der Initialisierung in Kauf nehmen zu müssen. Mit Optimierungen könnte das Problem jedoch auch nicht unter linearer Zeit gelöst werden, da zum vollständigen Lesen der Markow-Kette mindestens über die Kanten iteriert werden muss.

## 5 Ein Blick auf MDPs

Markow-Entscheidungsprozesse (*Markov-Decision-Processes, MDPs*) können als Erweiterung bzw. Verallgemeinerung von Markow-Ketten angesehen werden. Wir möchten daher in diesem Abschnitt einen kurzen Blick auf MDPs werfen, das Problem minimaler Varianzen adressieren und einen Ansatz zu dessen Lösung diskutieren.

### 5.1 Was sind MDPs?

**Definition 5.1** Ein Tupel  $M = (Q, A, P, I)$  nennen wir Markow-Entscheidungsprozess, wenn es folgende Eigenschaften erfüllt:

- (a)  $Q$  ist eine endliche Menge, d.h.  $|Q| < \infty$ .
- (b)  $A$  ist eine endliche Menge, genannt die Menge der Aktionen:  $|A| < \infty$ .
- (c)  $P$  ist eine Funktion  $Q \times A \times Q \rightarrow [0, 1]$  mit

$$\forall q \in Q, \alpha \in A : \sum_{q' \in Q} P(q, \alpha, q') \in \{0, 1\} \quad (52)$$

und beschreibt die Wahrscheinlichkeiten der Zustandsübergänge.

- (d)  $I$  ist eine Funktion  $Q \rightarrow [0, 1]$  mit  $\sum_{q \in Q} I(q) = 1$  und beschreibt die initiale Verteilung.
- (e) In jedem Zustand ist mindestens eine Aktion verfügbar:

$$\forall q \in Q : 0 < |\text{Act}(q)| < \infty \quad (53)$$

Dabei bezeichnen wir für  $q \in Q$  mit  $\text{Act}(q) := \{\alpha \in A \mid \exists q' \in Q : P(q, \alpha, q') > 0\}$  die Menge der Aktionen, für die eine Wahrscheinlichkeitsverteilung für Folgezustände existiert.

Die Semantik eines MDP ist so zu verstehen, dass von einem Zustand  $q \in Q$  aus zunächst eine Aktion  $\alpha \in \text{Act}(q)$  gewählt werden muss. Danach erfolgt ein probabilistischer Zustandsübergang wie in einer Markow-Kette, definiert durch  $q' \mapsto P(q, \alpha, q')$ . Die initiale Verteilung  $I$  ordnet jedem Zustand wie bei Markow-Ketten die Wahrscheinlichkeit zu, dass ein Lauf in diesem Zustand startet. Das Auswählen einer Aktion in jedem Zustand kann durch einen sogenannten Scheduler beschrieben werden:

**Definition 5.2** Sei  $M = (Q, A, P, I)$  ein Markow-Entscheidungsprozess. Ein Scheduler für  $M$  ist eine Funktion  $\mathcal{S} : Q^+ \rightarrow A$  mit der Eigenschaft

$$\forall n \in \mathbb{N}_{>0}, \pi \in Q^n : \mathcal{S}(\pi) \in \text{Act}(\pi_{n-1}). \quad (54)$$

Dabei beschreibt  $\pi_{n-1}$  den letzten Eintrag des Tupels  $\pi$ . Mit  $Q^+$  bezeichnen wir die Menge aller nichtleeren Tupel von Elementen aus  $Q$ , also  $Q^+ := \bigcup_{n \in \mathbb{N}_{>0}} Q^n$ . In Worten sagt die Bedingung, der Scheduler darf immer nur eine Aktion wählen, für welche Zustandsübergänge vom Zustand  $\pi_{n-1}$  existieren. Der Scheduler ordnet jedem Pfadpräfix die zu wählende Aktion zu und löst auf diese Weise den Nichtdeterminismus eines Markow-Entscheidungsprozesses auf. Wir bezeichnen die Menge aller Scheduler für  $M$  mit  $\text{Schedulers}(M)$ .

Offenbar ergibt sich aus einem MDP  $M = (Q, A, P, I)$  und einem Scheduler  $\mathcal{S} : Q^+ \rightarrow A$  für  $M$  eine eindeutig bestimmte Markow-Kette, bezeichnet durch  $M_{\mathcal{S}}$ , indem die Aktionen gemäß des Schedulers gewählt werden. Eine recht offensichtliche Konstruktion von  $M_{\mathcal{S}}$  ist in [1] beschrieben. Ziel dieses Abschnittes soll es sein, uns der folgenden Problemstellung zu widmen:

*Gegeben sei ein Markow-Entscheidungsprozess  $M = (Q, A, P, I)$ , eine Menge  $T \subseteq Q$  von Zielzuständen und eine Gewichtsfunktion  $R : Q \times A \times Q \rightarrow \mathbb{R}_{\geq 0}$  mit der Eigenschaft, dass unter jedem Scheduler für  $M$  in der resultierenden Markow-Kette mit sicherer Wahrscheinlichkeit ein Zustand aus  $T$  nach endlich vielen Schritten erreicht wird.*

*Gesucht ist ein Scheduler  $\mathcal{S}$  mit möglichst kleiner Varianz akkumulierter Kantengewichte in der resultierenden Markow-Kette  $M_{\mathcal{S}}$ . Gibt es für jeden MDP einen Scheduler mit minimaler Varianz oder existiert ein MDP mit einem Infimum der Varianz, wobei kein Scheduler das Infimum erreicht? Wie berechnen wir*

$$\inf_{\mathcal{S} \in \text{Schedulers}(M)} \text{Var}(M_{\mathcal{S}})?$$

## 5.2 Resultierende Markow-Ketten

Die Problemstellung möchten wir in erster Linie als Motivation verstehen. Wir werden die Fragen nicht abschließend beantworten, jedoch möchten wir einen Ansatz präsentieren und begründen, wie zum Finden von Schemulern, welche eine kleine Varianz aufweisen, vorgegangen werden kann. Dazu beginnen wir, die bereits genannte resultierende Markow-Kette  $M_{\mathcal{S}}$  formal zu präzisieren. Jedoch werden wir eine andere Konstruktion verwenden als Baier und Katoen [1]. Zunächst möchten wir eine dafür nützliche Schreibweise einführen:

**Definition 5.3** Seien  $M = (Q, A, P, I)$  ein MDP,  $\mathcal{S} \in \text{Schedulers}(M)$  ein Scheduler für  $M$  und  $q \in Q$  ein beliebiger Zustand von  $M$ . Dann definieren wir:

$$\mathcal{S}_{\leftarrow q} : Q^+ \rightarrow A : (p_0, \dots, p_{n-1}) \mapsto \mathcal{S}(q, p_0, \dots, p_{n-1}) \quad (55)$$

$\mathcal{S}_{\leftarrow q}$  ist offensichtlich wieder ein Scheduler für  $M$ :  $\mathcal{S}_{\leftarrow q} \in \text{Schedulers}(M)$

Wir beschreiben nun eine rudimentäre Art Markow-Kette, die wir generische Markow-Kette nennen wollen.

**Definition 5.4** Eine Tupel  $G = (Q, P)$  heißt generische Markow-Kette, wenn eine Markow-Kette  $M$  mit  $M = (Q, P, I)$  existiert. Eine generische Markow-Kette ist so gesehen eine Markow-Kette abzüglich der initialen Verteilung.

Wir wollen diese Definition nun nutzen, um jedem MDP  $M$  eine generische Markow-Kette zuzuordnen, welche durch Hinzufügen einer jeweils geeigneten initialen Verteilung zur resultierenden Markow-Kette  $M_S$  für einen beliebigen Scheduler  $S \in \text{Schedulers}(M)$  komplettiert werden kann.

**Definition 5.5** Sei  $M = (Q, A, P, I)$  ein MDP. Mit

$$\text{Gen}(M) := (\text{Schedulers}(M) \times Q, \tilde{P}) \quad (56)$$

bezeichnen wir die generische Markow-Kette zu  $M$ . Dabei sei  $\tilde{P}$  definiert als:

$$\begin{aligned} \tilde{P} : (\text{Schedulers}(M) \times Q) \times (\text{Schedulers}(M) \times Q) &\rightarrow [0, 1], \text{ mit} \\ ((\mathcal{S}_1, q), (\mathcal{S}_2, q')) &\mapsto \begin{cases} P(q, \alpha, q') & \text{falls } \mathcal{S}_2 = (\mathcal{S}_1)_{\leftarrow q}, \alpha = \mathcal{S}_1(q) \\ 0 & \text{sonst} \end{cases} \end{aligned} \quad (57)$$

Die generische Markow-Kette  $\text{Gen}(M)$  ist wohldefiniert, denn  $\mathcal{S}_1(q) \in \text{Act}(q)$ . Da  $\text{Act}(q)$  mindestens eine Aktion beinhaltet, gibt es für jeden Zustand der generischen Markow-Kette eine Menge ausgehender Kanten, deren Wahrscheinlichkeiten sich aufgrund der Bedingungen aus Definition 5.1 zu 1 aufsummieren.

Für einen MDP  $M = (Q, A, P, I)$  und einen Scheduler  $\mathcal{S} \in \text{Schedulers}(M)$  lässt sich die resultierende Markow-Kette nun auffassen als  $M_S := (Q', P', I')$  mit  $(Q', P') = \text{Gen}(M)$  und

$$I'(\mathcal{S}', q) := \begin{cases} I(q) & \text{falls } \mathcal{S}' = \mathcal{S} \\ 0 & \text{sonst} \end{cases}. \quad (58)$$

Obleich diese Markow-Kette in vielen Fällen zahlreiche unerreichbare Zustände mit sich bringt, welche für z.B. die Betrachtung von Varianzen irrelevant sind, erhalten wir zu einer Familie von Schemulern stets eine Familie von Markow-Ketten, die sich höchstens in der initialen Verteilung unterscheiden.

Wir wollen nichtnegative Kantengewichte auf Markow-Entscheidungsprozessen analog zu Kantengewichten auf Markow-Ketten betrachten. Solche Kantengewichte beschreiben wir durch eine Funktion

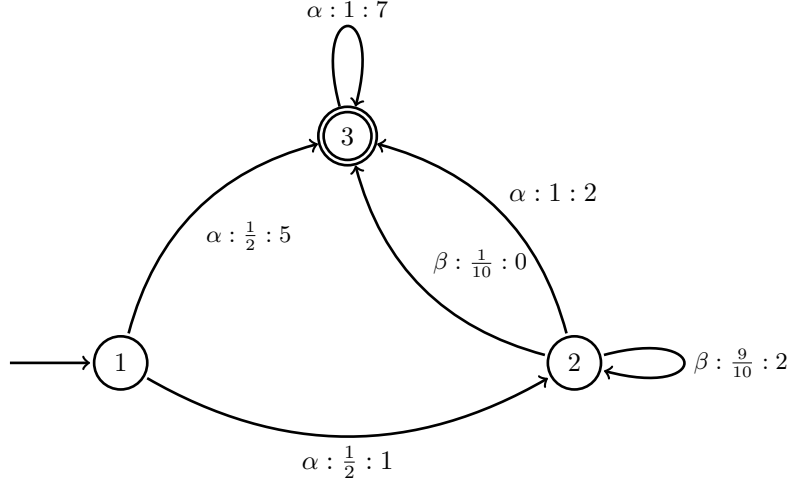
$$R : Q \times A \times Q \rightarrow \mathbb{R}_{\geq 0}. \quad (59)$$

Wenn von Zustand  $q \in Q$  mit Wahl von Aktion  $\alpha \in A$  in Zustand  $q' \in Q$  übergegangen wird, so wird das Gewicht  $R(q, \alpha, q')$  aufgesammelt. Wir wollen kurz darauf eingehen, wie die Kantengewichte in der resultierenden Markow-Kette aussehen: Für einen MDP  $M = (Q, A, P, I)$  und  $\mathcal{S} \in \text{Schedulers}(M)$  sowie  $q, q' \in Q$  ergibt sich für die Gewichtsfunktion  $\tilde{R}$  der resultierenden Markow-Kette

$$\tilde{R}((\mathcal{S}, q), (\mathcal{S}_{\leftarrow q}, q')) = R(q, \mathcal{S}(q), q'). \quad (60)$$

Während beispielsweise maximale Wahrscheinlichkeiten zum Erreichen einer Zielmenge stets durch einen sogenannten memoryless Scheduler erreicht werden können [1], d.h. einen Scheduler, der in einem Zustand unabhängig von dem vorangegangenen Pfadpräfix immer dieselbe Aktion wählt, stellen wir zunächst fest, dass diese Art von Scheduler für unser Problem nicht ausreichend ist. Dies können wir anhand eines einfachen Beispiels zeigen:

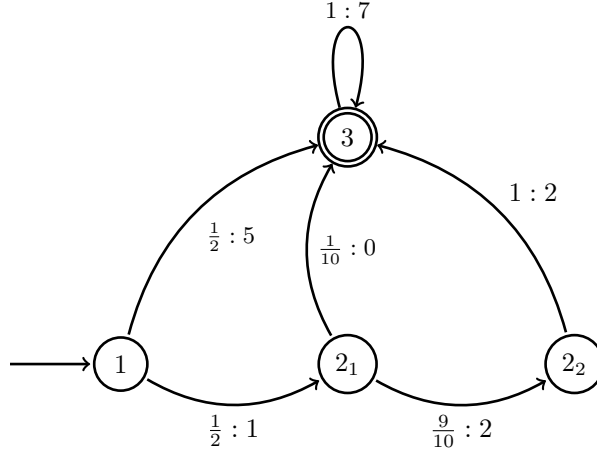
**Beispiel 5.6** Wir verwenden hier eine Notation analog zu den Darstellungen für Markow-Ketten, siehe Beispiel 2.18. Kanten werden in der Form  $\alpha : p : r$  beschriftet, wobei eine solche Kante von  $q_1$  nach  $q_2$  bedeutet, dass  $\alpha \in \text{Act}(q_1)$ ,  $P(q_1, \alpha, q_2) = p$  und  $R(q_1, \alpha, q_2) = r$  gilt:



Für Beispiel 5.6 gibt es genau zwei memoryless Scheduler, da nur die Wahl zwischen  $\alpha$  und  $\beta$  im Zustand 2 besteht:

- Wählen wir  $\alpha$ , so ergibt sich als Erwartungswert in der resultierenden Markow-Kette  $\mathcal{E}(R) = 4$  und als Varianz  $\mathcal{V}ar(R) = 1$  bei genau zwei möglichen Pfaden mit einer Wahrscheinlichkeit von je  $\frac{1}{2}$ .
- Wählen wir jedoch  $\beta$ , so ergibt sich nach  $\mathcal{E}_2(R) = \frac{9}{10} \cdot (\mathcal{E}_2(R) + 2) + \frac{1}{10} \cdot 0$  schließlich  $\mathcal{E}_2(R) = 18$  und  $\mathcal{E}_1(R) = 12$ . Analog zu unseren Betrachtungen aus vorangegangenen Abschnitten, bezeichnen wir den Erwartungswert aufsummierter Gewichte aus der Funktion  $R$  für den Start in Knoten  $i$  mit  $\mathcal{E}_i(R)$ . Die Varianz in diesem Fall ergibt sich zu  $\mathcal{V}ar(R) = \frac{1}{2} \cdot (12 - 5)^2 + \dots \geq \frac{49}{2}$  und wir betrachten diesen Scheduler nicht weiter, da wir bereits zuvor einen besseren (bzgl. Minimierung der Varianz) gesehen haben.

Betrachten wir nun einen Scheduler, der nicht zur Klasse der memoryless Scheduler gehört: Wählen wir beim ersten Besuch von Zustand 2 die Aktion  $\beta$  und beim zweiten Besuch von Zustand 2, sofern er denn eintritt, die Aktion  $\alpha$ , so erhalten wir eine Markow-Kette äquivalent zur folgenden Abbildung und mit den resultierenden Kenngrößen  $\mathcal{E}(R) = 4,8$  und  $\mathcal{V}ar(R) = 0,76$ .



Wir haben in der Abbildung bewusst nicht explizit unsere Definition einer resultierenden Markow-Kette aus einem MDP und einem Scheduler angewandt, sondern unerreichbare Zustände sofort außen vor gelassen. Mit diesem Beispiel haben wir gezeigt, dass die Klasse der memoryless Scheduler im Allgemeinen nicht zu minimalen Varianzen führt. Gleichzeitig wollen wir noch einmal darauf hinweisen, dass an dieser Stelle noch nicht geklärt ist, ob tatsächlich für jeden MDP ein Scheduler mit minimaler Varianz existiert oder ob nur eine Folge von Schemulern existieren muss, sodass der Grenzwert der Varianzen gegen das Infimum der Varianzen aller Scheduler konvergiert.

### 5.3 Ein Ansatz zum Minimieren von Varianzen

Als Intuition aus Beispiel 5.6 können wir uns überlegen: Nehmen wir einmal an, wir kennen den Erwartungswert, zu welchem ein Scheduler mit minimaler oder zumindest möglichst kleiner (nah am Infimum) Varianz existiert. Dann können wir versuchen, den Scheduler so zu konstruieren, dass wir auf jedem Pfad die Summe der Kantengewichte möglichst nah an den bekannten Erwartungswert bringen. Wir wollen aus dieser Intuition einen Ansatz entwickeln, um Scheduler mit minimalen oder zumindest ziemlich kleinen Varianzen zu ermitteln.

Wir definierten bereits die Varianz für Markow-Ketten. Wir können nun die Varianz akkumulierter Kantengewichte bis zum Erreichen einer Zielzustandsmenge  $T$  in einem MDP  $M = (Q, A, P, I)$  in Abhängigkeit der Gewichtsfunktion  $R$  und des Schedulers  $\mathcal{S}$  schreiben als

$$\text{Var}^{\mathcal{S}}(R) = \sum_{q \in Q} I(q) \cdot \sum_{\pi \in \text{Paths}_{(\mathcal{S}, q) \rightarrow (\text{Schedulers}(M) \times T)}(\text{Gen}(M))} \tilde{P}(\pi) \cdot (\tilde{R}(\pi) - \mu^{\mathcal{S}})^2,$$

wenn  $\text{Gen}(M) = (\text{Schedulers}(M) \times Q, \tilde{P})$  und  $\tilde{R}$  die entsprechende Gewichtsfunktion auf der resultierenden Markow-Kette ist. Dies entspricht unserer bisherigen Definition von Varianz und bezieht stets den Erwartungswert

$$\mu^{\mathcal{S}} = \sum_{q \in Q} I(q) \cdot \sum_{\pi \in \text{Paths}_{(\mathcal{S}, q) \rightarrow (\text{Schedulers}(M) \times T)}(\text{Gen}(M))} \tilde{P}(\pi) \cdot \tilde{R}(\pi)$$

ein. Wir nutzen jeweils wieder die übliche Fortsetzung von Wahrscheinlichkeiten bzw. Kantengewichten auf Pfade. Nun wollen wir einen im Vergleich zur Varianz allgemeineren Begriff einführen:

**Definition 5.7 (Target-Varianz)** Seien  $(\Omega, P)$  ein diskreter Wahrscheinlichkeitsraum,  $X$  eine Zufallsvariable auf  $(\Omega, P)$  und  $t \in \mathbb{R}$ . Mit

$$\mathcal{VarTar}_{(\Omega, P)}(X, t) := \mathcal{E}_{(\Omega, P)} \left( (X - t)^2 \right) \quad (61)$$

bezeichnen wir die Target-Varianz von  $X$  zum Ziel  $t$  auf  $(\Omega, P)$ . Sollte der Kontext  $(\Omega, P)$  klar sein, schreiben wir kurz  $\mathcal{VarTar}(X, t)$ .

Demnach betrachten wir nun die nach Wahrscheinlichkeiten gewichtete mittlere quadratische Abweichung zu einem frei gewählten Zielwert (*engl. target*)  $t$  statt zum Erwartungswert. Offenbar ist die Varianz ein Spezialfall der Target-Varianz, da  $\mathcal{VarTar}_{(\Omega, P)}(X, \mu) = \mathcal{Var}_{(\Omega, P)}(X)$ , wenn  $\mu = \mathcal{E}_{(\Omega, P)}(X)$  der Erwartungswert ist. Weiterhin stellen wir fest, dass sich die Target-Varianz als quadratische Funktion auffassen lässt:

**Lemma 5.8** *Sei  $(\Omega, P)$  ein diskreter Wahrscheinlichkeitsraum,  $X$  eine Zufallsvariable auf  $(\Omega, P)$  und  $t \in \mathbb{R}$ . Dann gilt:*

$$\mathcal{VarTar}(X, t) = (t - \mathcal{E}(X))^2 + \mathcal{Var}(X) \quad (62)$$

*Beweis.*

$$\begin{aligned} \mathcal{VarTar}(X, t) &= \sum_{\omega \in \Omega} P(\omega) \cdot ((X(\omega) - t)^2) \\ &= \sum_{\omega \in \Omega} P(\omega) \cdot (t^2 - 2tX(\omega) + (\mathcal{E}(X))^2 - (\mathcal{E}(X))^2 + (X(\omega))^2) \\ &= (t - \mathcal{E}(X))^2 - (\mathcal{E}(X))^2 + \mathcal{E}(X^2) \\ &= (t - \mathcal{E}(X))^2 + \mathcal{Var}(X) \end{aligned}$$

□

Insbesondere sehen wir als unmittelbare Konsequenz von Lemma 5.8, dass die Target-Varianz nie kleiner sein kann als die Varianz:

**Korollar 5.9** *Sei  $(\Omega, P)$  ein diskreter Wahrscheinlichkeitsraum,  $X$  eine Zufallsvariable auf  $(\Omega, P)$  und  $t \in \mathbb{R}$ . Dann gilt:*

$$\mathcal{VarTar}(X, t) \geq \mathcal{Var}(X) \quad (63)$$

Wir haben nun gesehen, dass die Funktion

$$f : \mathbb{R} \rightarrow \mathbb{R} : t \mapsto \mathcal{VarTar}(X, t)$$

als quadratische Funktion der Form  $f(t) = (t - a)^2 + b$  mit  $a, b \in \mathbb{R}$  aufgefasst werden kann, wenn der diskrete Wahrscheinlichkeitsraum und  $X$  als konstant angesehen werden. Dies werden wir uns zu Nutze machen, um gezielt nach den Erwartungswerten zu suchen, für welche ein Scheduler zu einem MDP existiert, der eine Markow-Kette mit genau diesem Erwartungswert erzeugt und deren Varianz möglichst klein oder sogar minimal ist.

Für unseren Anwendungsfall wollen wir die Target-Varianz einer resultierenden Markow-Kette  $M_S$  für  $t \in \mathbb{R}$  und eine Kantenbewertung  $R$  wie folgt notieren:

$$\mathcal{VarTar}^S(R, t) = \sum_{q \in Q} I(q) \cdot \sum_{\pi \in \text{Paths}_{(S, q) \rightarrow (\text{Schedulers}(M) \times T)}(\text{Gen}(M))} P(\pi) \cdot (R(\pi) - t)^2 \quad (64)$$

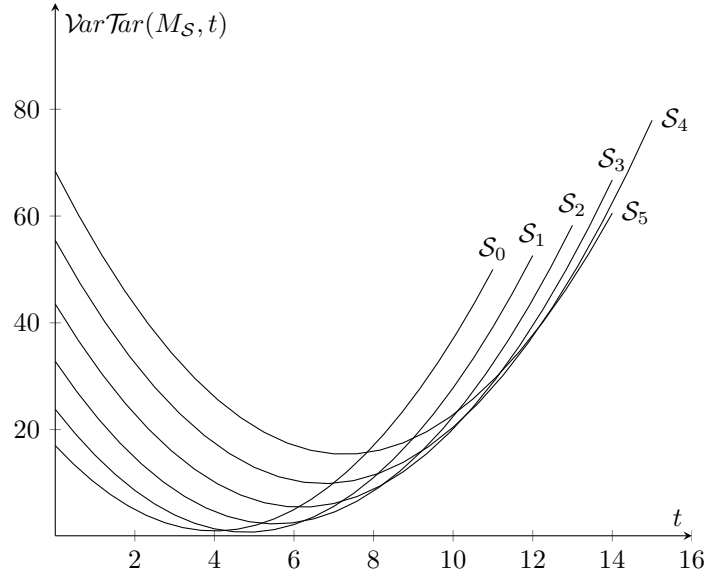
Nach vorangegangenen Betrachtungen stellen wir leicht fest, dass wir das Infimum der Varianzen von allen Markow-Ketten eines MDP für entsprechende Scheduler mit Target-Varianzen ausdrücken können:

**Lemma 5.10** *Sei  $M$  ein MDP und  $R$  eine Gewichtsfunktion auf  $M$ . Dann gilt:*

$$\inf_{\mathcal{S} \in \text{Schedulers}(M)} \mathcal{Var}^{\mathcal{S}}(R) = \inf_{t \in \mathbb{R}_{\geq 0}} \inf_{\mathcal{S} \in \text{Schedulers}(M)} \mathcal{VarTar}^{\mathcal{S}}(R, t) \quad (65)$$

*Beweis.* Wegen Korollar 5.9 ist die rechte Seite der Gleichung nicht kleiner als die linke. Da Varianzen ein Spezialfall der Target-Varianzen sind, enthält die Menge, über welche auf der rechten Seite das Infimum gebildet wird bereits alle Werte, die auf der linken Seite auftreten. Somit ist die linke Seite der Gleichung keinesfalls kleiner als die rechte. Beide Richtungen zusammen begründen Gleichheit.  $\square$

Wir werden im nächsten Abschnitt noch genauer darauf eingehen, wie man zu einem gegebenen  $t$  den Scheduler mit minimaler Target-Varianz  $\mathcal{VarTar}^{\mathcal{S}}(R, t)$  finden kann. Nehmen wir einmal an, wir hätten einen Algorithmus, welcher uns zu einem gegebenen  $t$  den Scheduler ausgibt mit minimaler Target-Varianz  $\mathcal{VarTar}^{\mathcal{S}}(R, t)$ , oder zumindest einen Scheduler, der dem Infimum ziemlich nahe kommt. Dann erlauben uns unsere bisherigen Beobachtungen, systematisch an verschiedenen Stellen  $t$  nach guten Schedulingern zu suchen. Im folgenden Diagramm werden die Target-Varianz-Verläufe einiger Scheduler  $\mathcal{S}_n$  für Beispiel 5.6 veranschaulicht. Dabei meint  $\mathcal{S}_n$  immer genau den Scheduler, welcher  $n$  mal  $\beta$  wählt und anschließend  $\alpha$ , insofern probabilistisch immer wieder Zustand 2 erreicht wird.



Wir wollen mögliche Argumente zur systematischen Suche beispielhaft erläutern:

- Starten wir (zufällig gewählt) bei  $t_0 = 14$ , so erhalten wir Scheduler  $\mathcal{S}_5$ , da  $\mathcal{VarTar}(M_{\mathcal{S}_5}, t_0) \leq \mathcal{VarTar}(M_{\mathcal{S}_i}, t_0)$  für alle hier betrachteten Scheduler  $\mathcal{S}_i$ . Wir berechnen die resultierende Markow-Kette  $M_{\mathcal{S}_5}$  und erhalten Varianz und Erwartungswert für diesen Scheduler, die dem globalen Minimum der Target-Varianz-Funktion zu  $\mathcal{S}_5$  im Diagramm entsprechen, in dem Falle (7.3, 15.4).

- Wir wiederholen die Bestimmung des Schedulers mit minimaler Target-Varianz für  $t_1 = 7.3$  und erhalten den Scheduler  $\mathcal{S}_2$ . Dieser hat sein Minimum bei  $(t_2, 2.3)$  mit  $t_2 = 5.5$ . Dass  $t_2 < t_1$  gilt, ist im Allgemeinen immer der Fall, wenn wir von  $t_0$  nach  $t_1$  mit  $t_0 > t_1$  iterativ gesucht haben. Andernfalls hätten wir an der Stelle  $t_0$  einen anderen Scheduler mit kleinerer Target-Varianz finden müssen. Das Intervall  $[7.3, 14]$  können wir bereits für die Suche nach besseren Schemulern ausschließen. Keine in x- und y-Richtung verschobene Normalparabel kann auf einem Teil des Intervalls kleinere Funktionswerte aufweisen, als die Parabeln der Scheduler, welche wir bisher gefunden haben und ein Minimum aufweisen, dass kleiner ist, als alle Minima von Schemulern, welche wir schon gefunden haben.
- Fahren wir so fort, dann erhalten wir im nächsten Schritt Scheduler  $\mathcal{S}_1$  mit dessen Minimum  $(t_3, 0.76)$ , wobei  $t_3 = 4.8$  ist. Nachdem wir geprüft haben, dass es bei  $t_3$  keinen besseren Scheduler gibt, wissen wir, dass wir im Intervall  $[t_3, t_0]$  nicht mehr suchen müssen. Sei  $k := 14 + \sqrt{\text{VarTar}(\mathcal{M}_{\mathcal{S}_5}, 14) - 0.76}$ . Weiterhin können wir auf eine Suche im Intervall  $[14, k)$  verzichten. Gäbe es in diesem Intervall eine noch kleinere Varianz als 0.76 zu entdecken, so hätten wir am Anfang nicht Scheduler  $\mathcal{S}_5$  erhalten dürfen an der Stelle  $t_0$ . Sehr wohl ist es im Allgemeinen denkbar in besagtem Intervall eine kleinere Target-Varianz zu finden. Dann jedoch hat die Parabel des zugehörigen Scheduler ihr Minimum an einer Stelle größer  $k$ , weswegen der Scheduler später gefunden wird. Als eine nächste Stelle zur weiteren Suche empfiehlt sich also  $k$ .

Das Vorgehen soll hier nur, wie geschehen, am Beispiel illustriert werden. Eine genaue algorithmische Präzisierung wollen wir hier nicht vornehmen, lediglich auf das Prinzip hinweisen. Wir haben gesehen, dass es recht leicht ist, systematisch zu suchen.

## 5.4 Minimieren von Target-Varianzen

Wir möchten uns nun kurz dem Problem widmen, für eine gegebene Stelle  $t$  den Scheduler  $\mathcal{S}$  zu finden, sodass  $\text{VarTar}^{\mathcal{S}}(R, t)$  minimal wird. Betrachten wir einen konkreten MDP  $M = (Q, A, P, I)$ , eine konkrete Zielmenge  $T \subseteq Q$ , dann schreiben wir die Target-Varianz von  $\tilde{R}$  zum Ziel  $t$  für den Start in  $(\mathcal{S}, q) \in \tilde{Q}$  in der generischen Markow-Kette  $\text{Gen}(M) = (\tilde{Q}, \tilde{P})$  als  $\text{VarTar}_{(\mathcal{S}, q)}(\tilde{R}, t)$ . Es gilt:

$$\text{VarTar}_{(\mathcal{S}, q)}(\tilde{R}, t) = \sum_{\pi \in \text{Paths}_{(\mathcal{S}, q)} \rightarrow (\text{Schedulers}(M) \times T)(\text{Gen}(M))} \tilde{P}(\pi) \cdot (\tilde{R}(\pi) - t)^2$$

Vergleichen wir mit Gleichung (64), so gilt:

$$\text{VarTar}^{\mathcal{S}}(R, t) = \sum_{q \in Q} I(q) \cdot \text{VarTar}_{(\mathcal{S}, q)}(\tilde{R}, t) \quad (66)$$

Wir erhalten unmittelbar folgendes Lemma:

**Lemma 5.11** *Sei  $\tilde{q} \in \tilde{Q} \setminus (\text{Schedulers}(M) \times T)$  kein Zielzustand. Dann gilt:*

$$\text{VarTar}_{\tilde{q}}(\tilde{R}, t) = \sum_{\tilde{q}' \in \tilde{Q}} P(\tilde{q}, \tilde{q}') \cdot \text{VarTar}_{\tilde{q}'}(\tilde{R}, t - \tilde{R}(\tilde{q}, \tilde{q}')) \quad (67)$$



*Beweis.* Sei  $\tilde{T} := \text{Schedulers}(M) \times T$ . Wir können den ersten Zustandsübergang des Pfades gesondert betrachten und erhalten:

$$\begin{aligned}
& \mathcal{VarTar}_{\tilde{q}}(\tilde{R}, t) \\
&= \sum_{\tilde{q}' \in \tilde{Q}} \cdot \sum_{\pi' \in \text{Paths}_{\tilde{q}' \rightarrow \tilde{T}}(\text{Gen}(M))} \tilde{P}(\tilde{q}, \tilde{q}') \cdot \tilde{P}(\pi') \cdot (\tilde{R}(\tilde{q}, \tilde{q}') + \tilde{R}(\pi') - t)^2 \\
&= \sum_{\tilde{q}' \in \tilde{Q}} \tilde{P}(\tilde{q}, \tilde{q}') \cdot \sum_{\pi' \in \text{Paths}_{\tilde{q}' \rightarrow \tilde{T}}(\text{Gen}(M))} \tilde{P}(\pi') \cdot (\tilde{R}(\pi') - (t - \tilde{R}(\tilde{q}, \tilde{q}')))^2 \\
&= \sum_{\tilde{q}' \in \tilde{Q}} \tilde{P}(\tilde{q}, \tilde{q}') \cdot \mathcal{VarTar}_{\tilde{q}'}(\tilde{R}, t - \tilde{R}(\tilde{q}, \tilde{q}')) \quad \square
\end{aligned}$$

Ist dagegen  $\tilde{q} \in (\text{Schedulers}(M) \times T)$  ein Zielzustand, so ist nach Definition 5.7  $\mathcal{VarTar}_{\tilde{q}}(\tilde{R}, t) = t^2$ . Definieren wir uns eine Funktion

$$g : Q \times \mathbb{R} \rightarrow \mathbb{R} : (q, t) \mapsto \inf_{\mathcal{S} \in \text{Schedulers}(M)} (\mathcal{VarTar}_{(\mathcal{S}, q)}(\tilde{R}, t)), \quad (68)$$

die jedem Zustand des MDP  $M = (Q, A, P, I)$  das Infimum aller Target-Varianzen für alle möglichen Scheduler zuordnet, so erhalten wir für nicht-Zielzustände  $q \notin T$

$$g(q, t) \quad (69)$$

$$= \inf_{\mathcal{S} \in \text{Schedulers}(M)} \mathcal{VarTar}_{(\mathcal{S}, q)}(\tilde{R}, t) \quad (70)$$

$$= \inf_{\mathcal{S} \in \text{Schedulers}(M)} \sum_{\tilde{q}' \in \tilde{Q}} \tilde{P}((\mathcal{S}, q), \tilde{q}') \cdot \mathcal{VarTar}_{\tilde{q}'}(\tilde{R}, t - \tilde{R}((\mathcal{S}, q), \tilde{q}')) \quad (71)$$

$$= \inf_{\mathcal{S} \in \text{Schedulers}(M)} \sum_{q' \in Q} P(q, \mathcal{S}(q), q') \cdot \mathcal{VarTar}_{(\mathcal{S} \leftarrow q, q')}(\tilde{R}, t - R(q, \mathcal{S}(q), q')) \quad (72)$$

$$= \inf_{\alpha \in \text{Act}(q)} \sum_{q' \in Q} P(q, \alpha, q') \cdot \inf_{\mathcal{S}' \in \text{Schedulers}(M)} \mathcal{VarTar}_{(\mathcal{S}', q')}(\tilde{R}, t - R(q, \alpha, q')) \quad (73)$$

$$= \min_{\alpha \in \text{Act}(q)} \sum_{q' \in Q} P(q, \alpha, q') \cdot g(q', t - R(q, \alpha, q')) \quad (74)$$

Zeile (71) erhalten wir durch Anwenden von Lemma 5.11. Anschließend können wir die Definitionen von  $\tilde{P}$  und  $\tilde{R}$  nutzen, um diese durch  $P$  und  $R$  auszudrücken und erhalten Zeile 72. Die Menge aller Scheduler können wir partitionieren nach der gewählten Aktion im anfänglichen Zustand  $q$ . Alle nach der Wahl von  $\alpha$  später aufzulösenden, nichtdeterministischen Entscheidungen zwischen Aktionen können wir in einem Scheduler  $\mathcal{S}'$  zusammenfassen. Auf diese Weise erhalten wir Zeile (73). Zum Schluss wenden wir erneut die Definition von  $g$  an. Da nach Definition 5.1 die Menge der wählbaren Aktionen endlich ist ( $|\text{Act}(q)| < \infty$ ), können wir das Infimum durch ein Minimum ersetzen. Für den trivialen Fall  $q \in T$  eines Zielzustandes erhalten wir einfach

$$g(q, t) = t^2. \quad (75)$$

Durch diese Betrachtung haben wir gezeigt, dass das folgende Infimum tatsächlich als Minimum existiert:

$$\inf_{\mathcal{S} \in \text{Schedulers}(M)} \mathcal{VarTar}_{(\mathcal{S}, q)}(\tilde{R}, t) = \min_{\mathcal{S} \in \text{Schedulers}(M)} \mathcal{VarTar}_{(\mathcal{S}, q)}(\tilde{R}, t) \quad (76)$$

Schließlich können wir jeweils anhand Zeile (74) eine Aktion wählen, die zum Minimum führt. Mit einer Funktion  $\text{sel}$ , definiert als

$$\text{sel} : Q \times \mathbb{R} \rightarrow A : (q, t) \mapsto \arg \min_{\alpha \in A} \sum_{q' \in Q} P(q, \alpha, q') \cdot g(q', t - R(q, \alpha, q')), \quad (77)$$

lässt sich ein optimaler Scheduler bezüglich Minimieren der Target-Varianz von  $R$  zum Ziel  $t$  eindeutig beschreiben: Sind wir im Zustand  $q$  mit dem Ziel  $t$ , dann wählen wir Aktion  $\alpha := \text{sel}(q, t)$  und ändern unser Ziel  $t$  entsprechend dem Kantengewicht  $r := R(q, \alpha, q')$  der probabilistisch gewählten Kante beim Übergang in den Folgezustand  $q'$  durch Subtraktion  $t \mapsto t - r$ . Sollte das argumentum minimi mehrdeutig sein, genügt es eine beliebige Aktion derer zu wählen, welche zum Minimum führen. Wir können den Scheduler formal definieren durch:

$$S_t(q) = \text{sel}(q, t) \quad (78)$$

$$S_t(q_0, q_1, \dots, q_n) = S_{t-R(q, S_t(q), q_1)}(q_1, \dots, q_n) \quad (79)$$

Wir zeigen kurz, dass dieser Scheduler tatsächlich optimal ist:

**Lemma 5.12**

$$\mathcal{VarTar}_{(S_t, q)}(\tilde{R}, t) = g(q, t) \quad (80)$$

*Beweis.* Offensichtlich gilt

$$\mathcal{VarTar}_{(S_t, q)}(\tilde{R}, t) \geq g(q, t),$$

da der Scheduler zu keiner kleineren Target-Varianz führen kann als zum Infimum.

Wir zeigen nun noch

$$\mathcal{VarTar}_{(S_t, q)}(\tilde{R}, t) \leq g(q, t). \quad (81)$$

Die Target-Varianzen von  $M_{S_t}$  berechnen sich per Definition durch die folgenden Gleichungen:

$\forall t \in \mathbb{R}, q \notin T :$

$$\mathcal{VarTar}_{(S_t, q)}(\tilde{R}, t) = \sum_{q' \in Q} P(q, S_t(q), q') \cdot \mathcal{VarTar}_{((S_t) \leftarrow q, q')}(\tilde{R}, t - R(q, S_t(q), q'))$$

$\forall t \in \mathbb{R}, q \in T :$

$$\mathcal{VarTar}_{(S_t, q)}(\tilde{R}, t) = t^2$$

Dabei können wir die Folge  $(\lambda_{(S_t, q)}(\tilde{R}, t)_n \mid n \in \mathbb{N})$  betrachten, die gegen  $\mathcal{VarTar}_{(S_t, q)}(\tilde{R}, t)$  konvergiert, gegeben durch:

$\forall t \in \mathbb{R}, q \in Q :$

$$\lambda_{(S_t, q)}(\tilde{R}, t)_0 := 0$$

$\forall t \in \mathbb{R}, q \notin T, i \in \mathbb{N}_{>0} :$

$$\lambda_{(S_t, q)}(\tilde{R}, t)_i := \sum_{q' \in Q} P(q, S_t(q), q') \cdot \lambda_{(S_t \leftarrow q, q')}(\tilde{R}, t - R(q, S_t(q), q'))_{i-1}$$

$\forall t \in \mathbb{R}, q \in T, i \in \mathbb{N}_{>0} :$

$$\lambda_{(S_t, q)}(\tilde{R}, t)_i := t^2$$

Wir erhalten Gleichung (81), weil  $\lim_{n \rightarrow \infty} \lambda_{(S_t, q)}(\tilde{R}, t)_n = \text{VarTar}_{(S_t, q)}(\tilde{R}, t)$  und da für alle  $n \in \mathbb{N}, q \in Q, t \in \mathbb{R}$  gilt:

$$\lambda_{(S_t, q)}(\tilde{R}, t)_n \leq g(q, t)$$

Dies zeigen wir per Induktion. Sei dazu  $n_0 \in \mathbb{N}$ . Wir nehmen an, die Ungleichung gilt für alle  $n < n_0$ . Falls  $n_0 = 0$ , so gilt, da Target-Varianzen immer nichtnegativ sind:

$$\lambda_{(S_t, q)}(\tilde{R}, t)_{n_0} = 0 \leq g(q, t)$$

Falls  $n_0 > 0$ ,  $q \in T$  gilt, so erhalten wir aufgrund von Gleichung (75)

$$\lambda_{(S_t, q)}(\tilde{R}, t)_{n_0} = t^2 = g(q, t).$$

Sei nun  $n_0 > 0$ ,  $q \notin T$ . Sei weiterhin  $t' := t - R(q, S_t(q), q')$ . Dann erhalten wir

$$\lambda_{(S_t, q)}(\tilde{R}, t)_{n_0} = \sum_{q' \in Q} P(q, S_t(q), q') \cdot \lambda_{((S_t) \leftarrow q, q')}(\tilde{R}, t - R(q, S_t(q), q'))_{n_0-1} \quad (82)$$

$$= \sum_{q' \in Q} P(q, S_t(q), q') \cdot \lambda_{(S_{t'}, q')}(\tilde{R}, t')_{n_0-1} \quad (83)$$

$$\leq \sum_{q' \in Q} P(q, S_t(q), q') \cdot g(q', t - R(q, S_t(q), q')) \quad (84)$$

$$= g(q, t) \quad (85)$$

□

Dabei erhalten wir Zeile (84) durch Anwenden der Induktionshypothese und Zeile (85) aufgrund von Gleichung (74).

Wie genau der optimale Scheduler bezüglich minimaler Target-Varianz berechnet werden kann, lassen wir hier offen. Jedoch könnte man für einen heuristischen Ansatz mit memoryless Schedulingern beginnen und diese iterativ verbessern, indem man für bestimmte Stellen  $t \in \mathbb{R}$  die zu wählende Aktion ändert, um einen besseren Scheduler zu erhalten. Wir haben bereits gesehen, dass der optimale Scheduler sich als Funktion  $S : Q \times \mathbb{R} \rightarrow A : (q, t) \mapsto \alpha$  auffassen lässt. Abschließend möchten wir noch eine Vermutung darlegen:

**Vermutung 5.13** *Für jeden MDP  $M = (Q, A, P, I)$  gibt es eine Zahl  $t_0 \in \mathbb{R}$  sodass für alle  $t_1, t_2 < t_0$  und alle Zustände  $q \in Q$  gilt:*

$$\text{sel}(q, t_1) = \text{sel}(q, t_2) \quad (86)$$

*Anders ausgedrückt wird vermutet, dass es genügt, wenn sich  $t$  unter einem MDP-spezifischen Schwellwert befindet, ein memoryless Scheduler hinreichend ist, um die optimale Aktion in jedem Zustand zu wählen.*

## 6 Zusammenfassung

Wir haben in dieser Arbeit zuerst einige Grundlagen der Wahrscheinlichkeitstheorie zusammengefasst. Im Anschluss beschrieben wir Markow-Ketten und

endliche Pfade in diesen, welche in einem bestimmten Knoten starten und in einem Knoten einer Zielzustandsmenge enden. Danach haben wir lineare Gleichungssysteme ermittelt, die eindeutig lösbar sind und deren Lösungen die Erwartungswerte, Varianzen bzw. Kovarianzen aufsummierter Kantengewichte auf den eben beschriebenen Pfaden sind. Wir haben die Methode der Nutzung linearer Gleichungssysteme zur Berechnung probabilistischer Kenngrößen implementiert und benötigte Rechenzeiten für die Ermittlung von Varianzen gemessen, mit dem Ergebnis, dass linear in der Kantenzahl der Markow-Kette die Zeit zur Berechnung der Varianzen wächst. Zum Schluss haben wir den Blick kurz auf Markow-Entscheidungsprozesse gelenkt und einen Ansatz betrachtet, wie man den Scheduler für einen MDP finden kann, dessen resultierende Markow-Kette eine möglichst kleine Varianz aufweist.

## 7 Appendix

Wir wollen nachfolgend solche mathematische Aussagen kurz darlegen und begründen, die nicht zentrales Thema dieser Arbeit sind, jedoch an bestimmten Stellen zur Argumentation herangezogen werden.

**Lemma 7.1** *Seien  $n \in \mathbb{N}$  eine natürliche Zahl und  $a \in \mathbb{R}_{>0} \setminus \{1\}$ . Dann gilt*

$$\sum_{i=0}^n a^i = \frac{a^{n+1} - 1}{a - 1} \quad (87)$$

*Beweis.*

$$(a - 1) \cdot \sum_{i=0}^n a^i = \sum_{i=0}^n a^{i+1} - \sum_{i=0}^n a^i = a^{n+1} - 1 \quad \square$$

**Korollar 7.2** *Betrachten wir den Grenzwert der Summe aus Lemma 7.1 für  $n \rightarrow \infty$ , so ergibt sich für  $0 < a < 1$ :*

$$\sum_{i=0}^{\infty} a^i = \lim_{n \rightarrow \infty} \sum_{i=0}^n a^i = \frac{1}{1 - a} \quad (88)$$

**Lemma 7.3** *Seien  $n \in \mathbb{N}$  eine natürliche Zahl und  $a \in \mathbb{R}_{>0} \setminus \{1\}$ . Dann gilt*

$$\sum_{i=0}^n i \cdot a^i = \frac{(an - n - 1)a^{n+1} + a}{(a - 1)^2} \quad (89)$$

*Beweis.*

$$\begin{aligned} (n+1)a^{n+1} + \sum_{i=0}^n i \cdot a^i &= \sum_{i=0}^{n+1} i \cdot a^i = \sum_{i=0}^n (i+1) \cdot a^{i+1} \\ &= a \sum_{i=0}^n i \cdot a^i + a \sum_{i=0}^n a^i = a \sum_{i=0}^n i \cdot a^i + a \cdot \frac{a^{n+1} - 1}{a - 1} \\ \Rightarrow (a-1) \sum_{i=0}^n i \cdot a^i &= (n+1)a^{n+1} - a \cdot \frac{a^{n+1} - 1}{a - 1} \quad \square \end{aligned}$$

**Korollar 7.4** Betrachten wir den Grenzwert der Summe aus Lemma 7.3 für  $n \rightarrow \infty$ , so ergibt sich für  $0 < a < 1$ :

$$\sum_{i=0}^{\infty} i \cdot a^i = \lim_{n \rightarrow \infty} \sum_{i=0}^n i \cdot a^i = \frac{a}{(a-1)^2} \quad (90)$$

**Lemma 7.5** Seien  $n \in \mathbb{N}$  eine natürliche Zahl und  $a \in \mathbb{R}_{>0} \setminus \{1\}$ . Dann gilt

$$\sum_{i=0}^n i^2 \cdot a^i = \frac{(-a^2n^2 + a(2n^2 + 2n + 1) - (n+1)^2)a^{n+1} + a(a+1)}{(a-1)^3} \quad (91)$$

*Beweis.*

$$\begin{aligned} n^2a^n + \sum_{i=0}^{n-1} i^2 \cdot a^i &= \sum_{i=0}^n i^2 \cdot a^i = \sum_{i=0}^{n-1} (i+1)^2 \cdot a^{i+1} \\ &= a \sum_{i=0}^{n-1} i^2 \cdot a^i + 2a \sum_{i=0}^{n-1} i \cdot a^i + a \sum_{i=0}^{n-1} a^i \\ &= a \left( \sum_{i=0}^n i^2 \cdot a^i - n^2a^n \right) + 2a \left( \frac{(an - a - n)a^n + a}{(a-1)^2} \right) + a \left( \frac{a^n - 1}{a-1} \right) \\ \Rightarrow (1-a) \sum_{i=0}^n i^2 \cdot a^i &= -n^2a^{n+1} + 2a \left( \frac{(an - a - n)a^n + a}{(a-1)^2} \right) + a \left( \frac{a^n - 1}{a-1} \right) \square \end{aligned}$$

**Korollar 7.6** Betrachten wir den Grenzwert der Summe aus Lemma 7.5 für  $n \rightarrow \infty$ , so ergibt sich für  $0 < a < 1$ :

$$\sum_{i=0}^{\infty} i^2 \cdot a^i = \lim_{n \rightarrow \infty} \sum_{i=0}^n i^2 \cdot a^i = \frac{a(a+1)}{(1-a)^3} \quad (92)$$

## Literatur

- [1] Christel Baier und Joost-Pieter Katoen. *Principles of Model Checking (Representation and Mind Series)*. The MIT Press, 2008. ISBN: 026202649X.
- [2] D. Demidov. „AMGCL: An Efficient, Flexible, and Extensible Algebraic Multigrid Implementation“. In: *Lobachevskii Journal of Mathematics* 40.5 (Mai 2019), S. 535–546. ISSN: 1818-9962. DOI: 10.1134/S1995080219050056. URL: <https://doi.org/10.1134/S1995080219050056>.
- [3] J.D. Dixon. „Exact Solution of Linear Equations Using P-Adic Expansions“. In: *Numerische Mathematik* 40 (1982), S. 137–142. URL: <http://eudml.org/doc/132821>.
- [4] T. Herman. „Probabilistic Self-stabilization“. In: *Information Processing Letters* 35.2 (1990), S. 63–67.
- [5] A. Itai und M. Rodeh. „Symmetry Breaking in Distributed Networks“. In: *Information and Computation* 88.1 (1990).
- [6] *MC Analyzer - a tool to calculate edge-reward expects, variances and co-variances in discrete time markov chains*. URL: [https://github.com/Necktschnagge/markov\\_chain\\_analyzer](https://github.com/Necktschnagge/markov_chain_analyzer).

- [7] *PRISM - Case Studies*. Jan. 2019. URL: <https://www.prismmodelchecker.org/casestudies/index.php>.
- [8] T. Verhoeff. „Reward variance in Markov chains : a calculational approach“. English. In: *Proceedings of the Eindhoven FASTAR Days 2004 (Eindhoven, The Netherlands, September 3-4, 2004)*. Hrsg. von L.G.W.A. Cleophas und B.W. Watson. Computer Science Reports. Technische Universiteit Eindhoven, 2004.