12 Dynamische Optimierung

Motivation/Einführung: Dynamische Optimierung
Sequentielle Optimierung
Stufenoptimierung

zwei mögliche Pfade

Für Planungsprobleme, deren Entscheidungsgrundlagen **problemseitig** nicht initial vollständig vorliegen,

- z.B. Lagerhaltung über mehrere Perioden
  $\Rightarrow$ stochastische Techniken fließen auf natürliche Weise ein
  (tatsächliche Unsicherheiten)

Für Planungsverfahren, welche Entscheidungen **verfahrensseitig** schrittweise treffen,

- z.B. Rucksack-Problem mit Entscheidungsbaum $\Rightarrow$
  randomisierte Techniken fließen künstlich ein
  (Konvergenzgründe, Effizienzgründe ...)

Ziele:

- Kennen lernen dynamisch zu optimierender Problemstellungen
- Erkennen warum schrittweise Optimierungsverfahren sinnvoll sein können
- Stochastische Einflüsse behandeln können
- Optimalitätskriterien für dynamische Probleme kennen lernen
- Lösungsmethoden der dynamischen Optimierung kennen lernen
- Markovsche Entscheidungsprozesse und deren Modellierungsmächtigkeit einordnen können
- Numerische Optimierungsmethoden für Markovsche Entscheidungsprozesse kennen lernen

Gliederung

12.1 Beispiele zur Einführung

Beispiel 1 Lagerhaltung:

Ein Gut sei zu lagern,

- über endlichem diskreten Planungszeitraum: n Perioden
- Belieferung jeweils zu Beginn einer Periode mit Lieferzugang $u_j \geq 0$, in Periode j (j=1,...,n)
- Auslieferung (gemäß Nachfrage) zu Beginn Periode unmittelbar nach Belieferung, sei $r_j \geq 0$, Nachfrage in Periode j (j=1,...,n)
  Annahme: Auslieferung = Nachfrage
- Lagerbestand x_j zu Beginn der Periode j, unmittelbar vor der Belieferung (j=1,...,n)
- Gemäß Lagerbilanzgleichung (dynamische Nebenbedingung) gilt $x_{j+1} = x_j + u_j - r_j$ (j= 1 ,..., n)
- Annahmen: $x_j \geq 0$ für j= 2, ..., n+1 und $x_1 = x_{n+1} = 0$

- anfallende Kosten:
  - Belieferung: $h_B(u_j)$ „fix + variabel“
  - Lagerung: $h_L(x_j)$ und
  - Gesamt $g_j(x_j,u_j) = h_L(x_j) + h_B(u_j)$
- Optimierungsaufgabe:
  - bestimme Liefermengen u_j (Entscheidungsvariable)
  - unter Einhaltung der Nebenbedingungen / Annahmen
  - derart dass Gesamtkosten minimal
    (bei gegebenen Nachfragen r_j => eigentlich kein dynamisches Problem)

Formal: $\min \sum_{j=1}^{n} \left(g_j(x_j,u_j) \right)$

udN $x_{j+1} = x_j + u_j - r_j$ für $j = 1,2,...,n$

$x_1 = x_{n+1} = 0$

$x_j \geq 0$ und $u_j \geq 0$ für $j = 1,2,...,n$

Beispiel 2 Instandhaltung:

Ersatz verschlissener Systemkomponenten

- zu früh: großer Wertverlust, zu spät: große Wartungskosten
- Endlicher Planungszeitraum: n Perioden
- Entscheidung zu Beginn der Periode
  - falls $u_j=1$ dann Ersetzung in Periode j für j = 1, ..., n
  - sonst $u_j=0$ und keine Ersetzung
- Resultat: Alter der Komponenten
  - zu Ende Periode j, unmittelbar vor Periode j+1
  - x_j enthält Alter der Periode j j=1,...,n
  - gemäß Zusammenhang (dynamische Nebenbedingung)
  - $x_{j+1} = x_j+1$ falls nicht ersetzt
  - $x_{j+1} = 1$ falls ersetzt
  - daher $x_{j+1} = x_j (1 - u_j) + 1$ für j = 1, ..., n
  - Annahme: $x_1 = 1$

- Anfallende Kosten
  - Ersatz: $h_E(x_j)$ Beschaffung - Restwert
  - Wartung: $h_W(x_j)$
  - Gesamt: $g(x_j,u_j)=u_j \cdot h_E(x_j)+(1-u_j)h_W(x_j)$
- Optimierungsaufgabe:
  - bestimme Ersatzzeitpunkte j (Entscheidungsvariable u_j)
  - unter Einhaltung der Nebenbedingungen / Annahmen

derart, dass Gesamtkosten minimal werden.

Formal:
$$\min \sum_{j=1}^{n} g(x_j,u_j)$$

udN
$$x_{j+1} = x_j(1-u_j)+1$$
$$x_1 = 1$$
$$x_j \in \{1,2,...,j\}$$
$$u_j \in \{0,1\} \qquad \text{für } j=1,2,...,n$$

Beispiel 3 Rucksack-Problem:

Zur Erinnerung

n Gegenstände mit Gewichten $a_j > 0$ $(j=1,...,n)$

und Werten $c_j > 0$ $(j=1,...,n)$

Maximalgewicht $A > 0$

gepackte Werte-Summe zu maximieren

Entscheidungsvariablen (hier)

$u_j=1$ Gegenstand j wird eingepackt

$u_j=0$ Gegenstand j wird nicht eingepackt

Formal:

$$\max \sum_{j=1}^{n} c_j u_j \quad \text{udN} \quad \sum_{j=1}^{n} a_j u_j \le A \ \text{ und } u_j \in \{0,1\} \ \text{ für } j = 1,2,...,n$$

„künstliche“ Dynamisierung:

- Gegenstände („irgendwie“) geordnet
- Einpacken (oder nicht) „in Stufen“
  - auf Basis des noch verfügbaren Gewichtsrestes x_j
  - zu Ende Periode j-1, unmittelbar vor Entscheidung u_j
- resultierend in Gewichtsresten gemäß Zusammenhang (dynamische Nebenbedingung) $x_{j+1} = x_j - a_j u_j$ (für j=1,...,n)
  initale Gewichtsreserve: $x_1 = A$

Formal:

$$\max \sum_{j=1}^{n} c_j u_j$$

udN

$$u_{j+1} \in \{0,1\}, \quad x_{j+1} = x_j - a_j u_j \text{ für } j = 1,2,...,n \quad \text{und}$$

$$x_1 = A, 0 \le x_{j+1}$$

Beispiel 4: Maschinenbelegungsproblem

- Ausführung von 4 Operationen (A, B, C, D)
- Eine Maschine ist verfügbar
- Reihenfolgerestriktionen A muss vor B und D muss nach B durchgeführt werden
- Initiale Einrichtung der Maschine um Operation A oder C durchzuführen: S_A bzw. S_C
- Umrüstkosten von x nach y ($x,y \in \{A,B,C,D\}$): K_{xy}

Ziel: Finde eine Reihenfolge, die die Kosten minimiert!

Zustandsbeschreibung in Stufe i Vektor x der Länge i

$$x_0 = \emptyset,\ x_{j+1} = (x_j(1), \ldots, x_j(j), u_{j+1})$$

Definition der Menge $U_j(x_j)$:

$$A \in U_{j+1}(x_j) \quad \text{falls } \forall i = 1, \ldots, j : x_j(i) \neq A$$

$$B \in U_{j+1}(x_j) \quad \text{falls } \forall i = 1, \ldots, j : x_j(i) \neq B \wedge \exists i = 1, \ldots, j : x_j(i) = A$$

$$C \in U_{j+1}(x_j) \quad \text{falls } \forall i = 1, \ldots, j : x_j(i) \neq C$$

$$D \in U_{j+1}(x_j) \quad \text{falls } \forall i = 1, \ldots, j : x_j(i) \neq D \wedge \exists i = 1, \ldots, j : x_j(i) = B$$

Zielfunktion: $$\min_{u_j \in U_j(x_j)} \left(S_{u_1} + \sum_{j=2}^{4} K_{x_j(j-1), u_j} \right)$$

12.2 Problemstellung der dynamischen Programmierung

Formalisierungen der Beispiele

- zwar sehr problemabhängig (kein Automatismus)
- aber weitgehend ähnlich

d.h.

- Systembetrachtung über endlichen Planungszeitraum,
  wird eingeteilt in Perioden / Stufen j (j=1,...,n)
- zu Beginn Periode j (mit Beendigung Periode j-1) ist System im Zustand x_j
  (Zustandsvariable)
- zu Beginn des Planungszeitraums herrscht Anfangszustand $x_1=x_a$
- in Periode j wird Entscheidung u_j getroffen
  (u_j ist Entscheidungsvariable / Steuervariable),
- resultiert zu Folgezustand $x_{j+1}= f_j(x_j,u_j)$
  (abhängig vom Vorzustand und Entscheidung)
- und Kosten / Erlösen (rewards) $g_j(x_j,u_j)$
  (abhängig vom Vorzustand und Entscheidung)

Restriktionen bestehen (potenziell)

- in Form erlaubter (nichtleerer) Steuerbereiche $U_j(x_j)$, $u_j \in U_j(x_j) \neq \emptyset$
- in Form erlaubter (nichtleerer) Zustandsbereiche $x_{j+1} \in X_{j+1} \neq \emptyset$ wobei $X_1 = \{x_1\}$
- Steuerbereiche $U_j(x_j)$ und Zustandsbereiche X_{j+1} fallabhängig eindimensional / mehrdimensional, reellwertig / ganzzahlig / zweiwertig, ...
- Funktionen f_j, g_j definiert auf $D_j := \{(x,u) | \, x \in X_j, \, u \in U_j(x)\}$
- Optimierungsaufgabe besteht aus Minimierung Kosten (Maximierung Erlöse) über gesamten Planungszeitraum

$$\min \sum_{j=1}^{n} g_j(x_j, u_j) \quad \text{udN} \quad x_{j+1} = f_j(x_j, u_j) \text{ für } j = 1,2,...,n \text{ und } x_1 = x_a$$
$$x_{j+1} \in X_{j+1} \text{ und } u_j \in U_j(x_j) \text{ für } j = 1,2,...,n$$

kann sehr komplex sein!

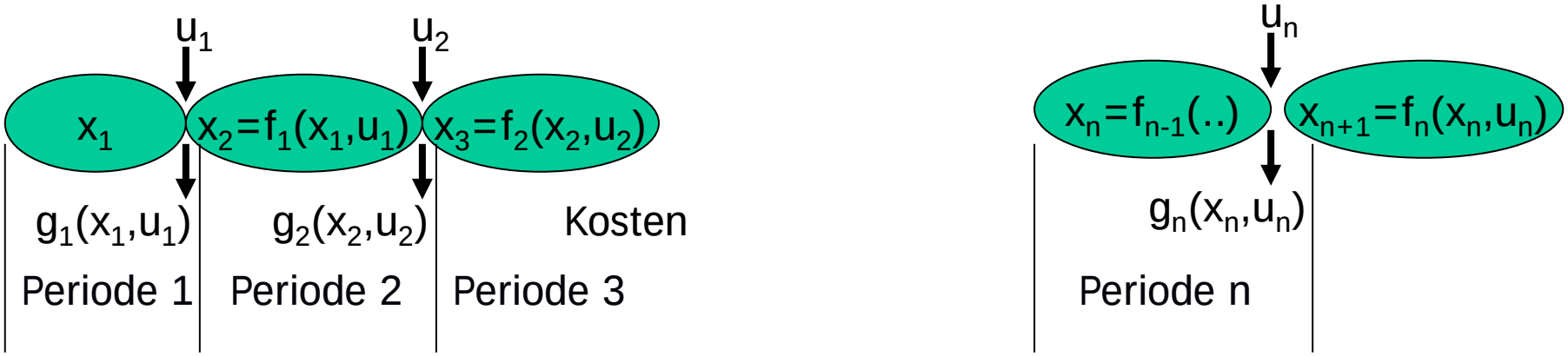
Struktur der vorgestellten Beispiele

	Lagerhaltung	Erneuerung	Rucksack	Maschinenbelegung
x_j X_j	reellwertig Bestand $\mathbb{R}_+$für $j = 2, \ldots, n$ $X_0 = X_{n+1} = \{0\}$	ganzzahlig Alter $\{1, \ldots, j-1\}$ für $j = 1, \ldots, n+1$, $X_1 = \{1\}$	reellw. Kapazität $[0, A]$ für $j = 2, \ldots, n+1$ $X_1 = \{A\}$	Vektor über $\{A..D\}$ Vektor der Länge j $X_0 = \emptyset$
u_j $U_j(x_j)$	reellwertig/ ganzzahlig	binär $\{0, 1\}$	binär $\{0, 1\}$	Element aus $\{A..D\}$ abhängig von x_j
$f_j(x_j, u_j)$ $g_j(x_j, u_j)$	$x_j + u_j - r_j$ $h_B(u_j) + h_L(x_j)$	$x_j(1-u_j)+1$ $u_j h_E(x_j)$ $+(1-u_j)h_W(x_j)$	$x_j - a_j u_j$ $c_j u_j$	$(x_j(1), \ldots, x_j(j), u_j)$ $g_{j-1}(x_{j-2}, u_{j-1})$ $+K_{u_{j-1}, u_j}$
	Kosten	Kosten	Nutzen (maximieren!)	Kosten

Problemstellung (wie beschrieben) deckt nicht den allgemeinsten Fall, da:

- f_j und g_j nur von Vorzustand und Entscheidung abhängig sind
  - wird in stochastischen Modellen „Markovsch“ heißen
  - ist durch Definition „Zustand“ immer erreichbar
- f_j und g_j deterministisch definiert
  - Änderung wird zu stochastischen Modellen führen
- „Zeit“ –Betrachtung eingeschränkt, da
  - in Perioden ablaufend (implizit: gleiche Länge?)
  - Zustandsübergänge „sprunghaft“ geschehen: „zeitdiskrete“ Modelle
  - beschränkt, endlich
  - ...

Veranschaulichung (Entscheidungszeitpunkte hierdurch nicht festgelegt)



Problemstellung verfolgt System über Zeit / über Stufen

- auf Basis einer Entscheidungsfolge (Politik / Steuerung) $u_1,...,u_n$
- welche zu (zugehöriger) Zustandsfolge $(x_1,...,x_n,x_{n+1})$ führt, wobei $x_1:=x_a$ und $x_{j+1} :=f_j(x_j,u_j)$ $j=1,...,n$

Eigenschaften

- zulässige Politik / Zustandsfolge
  - genügt Nebenbedingungen
  - generiert zulässige Lösung
- optimale Lösung / Politik / Zustandsfolge
  (Existenz vorausgesetzt)
  - ist zulässig
  - erreicht Optimierungsziel

Sind Funktionen f_j, g_j für j=1,...,n

und Zustands-, Steuerbereiche X_{j+1}, U_j für j=1, ..., n gegeben,

- dann hängt optimale Lösung - sofern existent – offensichtlich vom Anfangszustand x_1 ab
- daher Bezeichnung für dieses Gesamtproblem $P_1(x_1)$.

Analog hängt optimale Lösung des (Teil-)Problems,

welches nur Perioden j,...,n umfasst,

- ausschließlich von Anfangszustand x_j ab,
- daher Bezeichnung für dieses Teilproblem $P_j(x_j)$ für j=2,...,n

<u>Wesentliche Voraussetzung</u>:

- f_j, g_j nur von Vorzustand und Entscheidung abhängig! (aber Vergangenheit im Zustand kodierbar!)

Existiere für Teilproblem $P_j(x_j)$ eine optimale Politik $(u^*_j,u^*_{j+1},...,u^*_n)$ mit resultierendem Minimalwert der Zielfunktion $v^*_j(x_j)$

Aussage:

- Wird das Folgeproblem $P_{j+1}(x_{j+1})$ mit Anfangszustand
  $x_{j+1} = x^*_{j+1} := f_j(x_j,u^*_j)$ betrachtet,
  so ergibt sich für $P_{j+1}(x^*_{j+1})$ als optimale Politik $(u^*_{j+1},...,u^*_n)$
  mit Minimalwert (Teil-) Zielfunktion $v^*_{j+1}(x^*_{j+1})$

Beweisskizze:

- Wenn nicht, so gäbe es für $P_{j+1}(x^*_{j+1})$ bessere Politik $(u'_{j+1},...,u'_n)$ mit niedrigerem Wert $v'_{j+1}(x^*_{j+1})$
  und folglich auch für $P_j(x_j)$ bessere Politik mit $(u'_j, u'_{j+1},...,u'_n)$
  mit niedrigerem Wert
  $g_j(x_j ,u^*_j) + v'_{j+1}(x^*_{j+1}) < g_j(x_j ,u^*_j) + v^*_{j+1}(x^*_{j+1}) = v^*_j (x^*_j)$

=> Widerspruch zur Optimalitätsvoraussetzung für $v^*_j (x^*_j)$

Satz 12.1 (Bellmansches Optimalitätsprinzip)

Sei $(u^*_1,...,u^*_j,...,u^*_n)$ eine optimale Politik für $P_1(x_1)$ und x^*_j Zustand

(zu Beginn) Periode j, dann ist $(u^*_j,...,u^*_n)$ eine optimale Politik für $P_j(x^*_j)$

(d.h. optimale Folgeentscheidungen (ab j) sind unabhängig von Vorentscheidungen (vor j), allerdings abhängig von Anfangszustand x_j).

Der Beweis des Satzes folgt aus der Darstellung

$$v^*_j(x^*_j) = g_j(x^*_j, u^*_j) + v^*_{j+1}(x^*_{j+1}) \quad \text{und}$$

der Minimalität der beiden Wertefunktionen $v^*_j(x^*_j)$ und $v^*_{j+1}(x^*_{j+1})$.

Allgemeine Darstellung durch **Bellmansche Funktionsgleichung**:

$$v_j^*(x_j) = \min_{u \in U_j(x_j)} \{g_j(x_j, u_j) + v_{j+1}^*[f_j(x_j, u_j)]\}$$

mit $v^*_{n+1}(x_{n+1}) := 0$ für alle $x_{n+1} \in X_{n+1}$

(wesentlich Voraussetzung für alles Folgende: Existenz optimaler Lösungen)

Verschiedene Modifikationen sind einfach zu integrieren:

- Maximierung statt Minimierung durch Ersetzung von min durch max
- Anfangs- und Endzustand können aus einer Menge gewählt werden
  z.B. $v^*_1(x^*_1) = \min_{x1 \in X_1} v^*_1(x_1)$
  wobei X_1 Menge möglicher Anfangszustände
- Zielfunktion kann ein Produkt statt einer Summe sein
  (z.B. Zuverlässigkeit von Bauteilen in Serie)
  Bellmansche Funktionsgleichungen lauten dann

$$v_j^*(x_j) = \min_{u \in U_j(x_j)} \{ g_j(x_j, u_j) \cdot v_{j+1}^*[f_j(x_j, u_j)] \}$$

12.3 Bellmansche Funktionsgleichungsmethode

In direkter Anwendung der B´schen Funktionalgleichung,

- rückwärts, also für j=n, n-1, ..., 1
  - startend mit $v^*_{n+1}(x_{n+1}) := 0$ und $x_{n+1} \in X_{n+1}$
  - erreicht man sukzessiv (und hält fest / speichert) $v^*_j(x_j)$ und $x_j \in X_j$ für j=n, n-1, ..., 1
  - wobei abschließend $v^*_1(x^*_1)$ das gesuchte Optimum ist.
  - wobei man jeweils zusätzlich die Minimalstelle $z^*_j(x_j)$, die beste Entscheidung u_j des Klammerausdrucks der Gleichung

$$\left\{ g_j(x_j, u_j) + v^*_{j+1}[f(x_j, u_j)] \right\}$$

festhält, so dass für die Minimalstelle gilt

$$g_j(x_j, z^*_j(x_j)) + v^*_{j+1}[f(x_j, z^*_j(x_j))]$$
$$= \min_{u_j \in U_j(x_j)} \left\{ g_j(x_j, u_j) + v^*_{j+1}[f(x_j, u_j)] \right\} = v^*_j(x_j)$$

Folge $(z^*_1,...,z^*_n)$ wird als **optimale Rückkopplungssteuerung** bezeichnet

Damit lässt sich „vorwärts“ für j=1,2,...,n

(in Verfolgung der Bestentscheidungen und der daraus resultierenden Zustände)

- die optimale Politik $(u^*_1,...,u^*_n)$
  und die optimale Zustandsfolge $(x^*_1,...,x^*_{n+1})$
  konstruieren, gemäß

$x^*_1=x_a \Rightarrow u^*_1 = z^*_1(x^*_1)$

$\rightarrow x^*_2 = f_1(x^*_1,u^*_1) \Rightarrow u^*_2=z^*_2(x^*_2)$

$\rightarrow ... \quad ... \Rightarrow u^*_n = z^*_n(x^*_n)$

$\rightarrow x^*_{n+1} = f_n(x^*_n,u^*_n)$

Die berechnete Politik $(u^*_1,...,u^*_n)$ und die Zustandsfolge $(x^*_1,...,x^*_{n+1})$ sind optimal für das Problem $P_1(x^*_1=x_a)$

Speicherung der Werte $v^*_j(x_j)$ und $z^*_j(x_j)$ für jeden Zustand x_j

Alternative Berechnung (Variante 2):

- Bestimme in der Rückwärtsberechnung nur die Funktionswerte $v^*_j(x_j)$ (nicht aber die optimalen Entscheidungen $z^*_j(x_j)$)
- Der optimale Funktionswert entspricht dem minimalen $v^*_1(x_1)$
- Bestimme in der Vorwärtsphase aus der Kenntnis der $v^*_j(x_j)$ die jeweils optimale Entscheidung, aus der sich der optimale Nachfolgezustand ergibt

- Beide Varianten bestehen aus einer Vorwärts- und einer Rückwärtsphase,
- die Rückwärtsphase durchläuft
  - alle Zustände und
  - alle Wege zwischen Zuständen der Phase j+1 und j (j=1,...,n-1)
- die Vorwärtsphase
  - durchläuft den optimalen Pfad

Algorithmus zur dynamischen Optimierung (Variante 1):

Schritt I (Rückwärtsrechnung)

$v^*_{n+1}(x_{n+1}) := 0$ for all $x_{n+1} \in X_{n+1}$

For j = n downto 1 do

For all $x_j \in X_j$ do

$$v^*_j(x_j) := \min_{u_j \in U_j(x_j)} \{g_j(x_j,u_j) + v^*_{j+1}(f_j(x_j,u_j))\} ;$$

$$z^*_j(x_j) := \operatorname{argmin}_{u_j \in U_j(x_j)} \{g_j(x_j,u_j) + v^*_{j+1}(f_j(x_j,u_j))\} ;$$

Schritt II (Vorwärtsrechnung)

$x^*_1 := x_a$;

For j = 1 to n do

$$x^*_{j+1} := f_j(x^*_j, z^*_j(x^*_j)) ;$$

Vorgehen erfordert keine speziellen Voraussetzungen an die Gestalt der Funktionen g_j und f_j!

Beispiel 1: Einfaches Lager- und Liefer-System

Einfaches Lager für einen Warentyp,

- das periodisch beliefert wird und aus dem periodisch ausgeliefert wird
- Planung / Verfolgung des Lagersystems für n = 4 Perioden
- beschränkte Lagerkapazität: max 2 „Stück“
- Lagerkosten: 0 vernachlässigt
- initiale Lagerbelegung: 0 leer
- beschränkte Belieferungskapazität: max 2 Stück / Periode
- Belieferungskosten saisonal schwankend, aber bekannt,
  - in Periode j: q_j € / Stück $q_1=7$, $q_2=9$, $q_3=12$, $q_4=10$
- feste Auslieferungsmengen 1 Stück / Periode
  - mit Auslieferung aus Lager oder direkt aus Anlieferung
  - Auslieferungskosten: 0 vernachlässigt

Gesucht ist optimale Politik

- d.h. Entscheidungsfolge bzgl Belieferungsmengen u_j Stück / Periode j
- die zu Lagerzuständen x_j Stück zu Beginn Periode j führt
- die die Nebenbedingungen wahrt und
- die minimale Kosten $u_1q_1+u_2q_2+u_3q_3+u_4q_4$ verursacht

Formalisierung:

- Steuerbereiche grob $U_j(x_j) \equiv \{0,1,2\}$ $(j=1,...,4)$
  könnten zustandsabhängig begrenzt werden durch volles Lager
- Zustandsbereiche grob $X_j=\{0,1,2\}$ $(j=1,...,4)$
  - davon abweichend begrenzt
    Initialzustand $X_1=\{0\}$
    sinnvoller Endzustand $X_5=\{0\}$

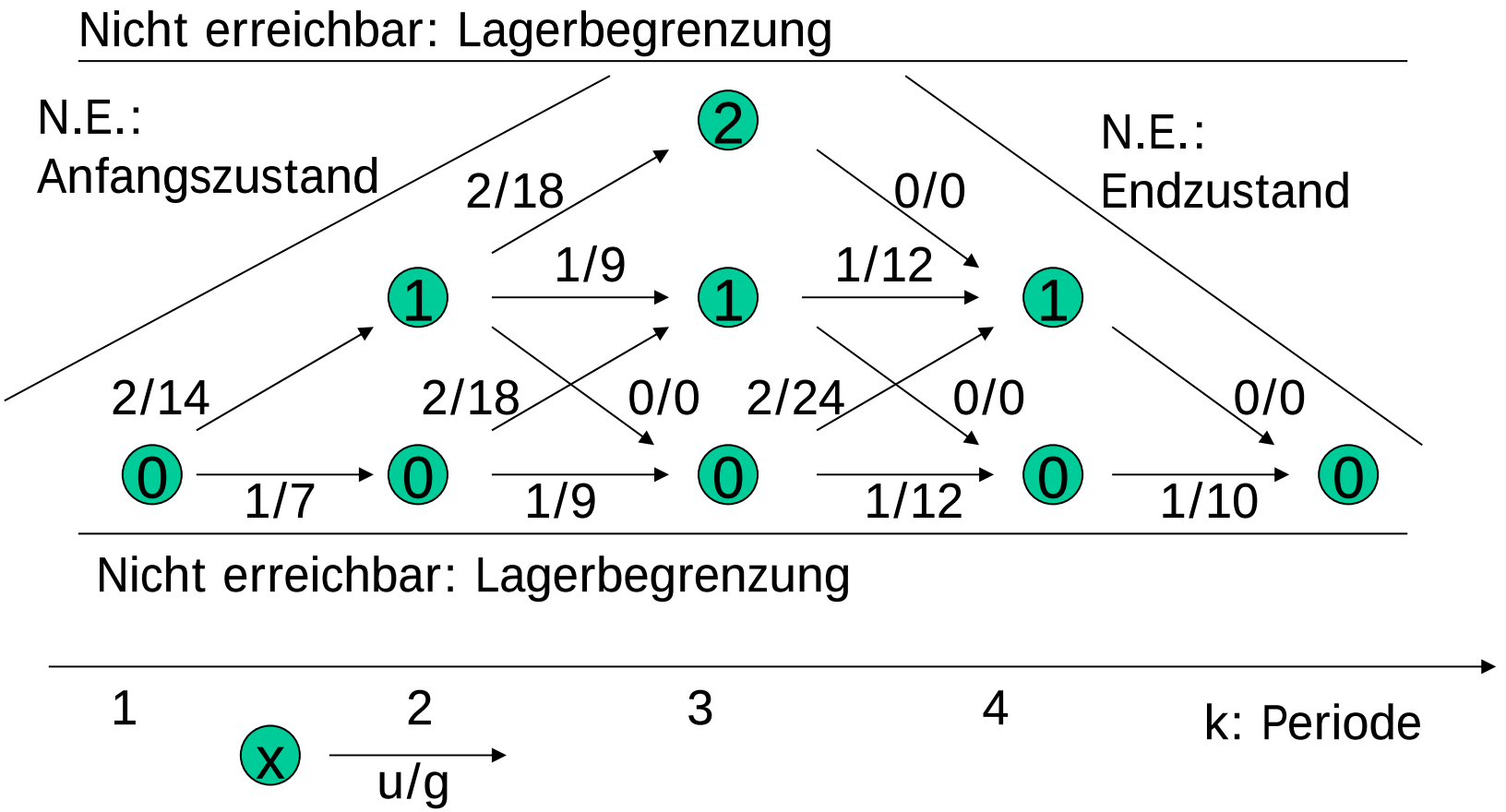
I.a. tritt ganz X als Menge möglicher Endzustände auf!
Randbed: $v^*_{n+1}(x_{n+1})=0$

- Zustands-(Transformations-)Funktionen
  $(x_{j+1}=)\quad f_j(x_j,u_j) = x_j + u_j - 1$ $(j=1,...,4)$
- Kostenfunktionen
  $g_j(x_j,u_j) = u_j\, q_j$ $(j=1,...,4)$

Zustandsbereiche endlich

- Veranschaulichung des Problemraums mittels bewerteter Graphen
- Darstellung der (diversen) Funktionen im Verfahrensverlauf mittels Tabellen

Veranschaulichung des Problemraums



Durchführung: Bellmansche Funktionalgleichungsmethode mit

Schlüsselbeziehung: $$v_j^*(x_j) = \min_{u_j \in U_j(x_j)} \left\{ g_j(x_j, u_j) + v_{j+1}^*[f(x_j, u_j)] \right\}$$

Rückwärtsentwicklung, Periode 4

x_4	u_4	$x_5=f_4(x_4,u_4)$	$v^*_5(x_5)$	$g_4(x_4,u_4)$	$v_4(x_4,u_4)$	$z^*_4(x_4)$
0	0	-1				
	1	0	0	$q_4=10$	10*	1
	2	1				
1	0	0	0	0	0*	0
	1	1				
	2	2				
2	0	1				
	1	2				
	2	3				

Lager muss bei Periode 5 leer sein!

also für Periode 4 zwei mögliche Zustände mit jeweils optimalen Politiken

unzulässiger Bereich

optimale Werte

Zur Erinnerung:

$g_j(x_j,u_j) = u_j\, q_j$

$q_1=7,\ q_2=9,\ q_3=12,\ q_4=10$

Rückwärtsentwicklung, Periode 3

x_3	u_3	$x_4=f_3(x_3,u_3)$	$v^*_4(x_4)$	$g_3(x_3,u_3)$	$v_3(x_3,u_3)$	$z^*_3(x_3)$
0	0	-1				
	1	0	10	$q_3=12$	22*	1
	2	1	0	$2q_3=24$	24	
1	0	0	10	0	10*	0
	1	1	0	$q_3=12$	12	
	2	2				
2	0	1	0	0	0*	0
	1	2				
	2	3				

Periode 4 erlaubt nur Zustände 0 und 1

unzulässiger Bereich

optimale Werte

Zur Erinnerung:

$g_j(x_j,u_j) = u_j\, q_j$

$q_1=7,\ q_2=9,\ q_3=12,\ q_4=10$

Rückwärtsentwicklung, Periode 2

x_2	u_2	$x_3=f_2(x_2,u_2)$	$v^*_3(x_3)$	$g_2(x_2,u_2)$	$v_2(x_2,u_2)$	$z^*_2(x_2)$
0	0	-1				
	1	0	22	$q_2=9$	31	
	2	1	10	$2q_2=18$	28*	2
1	0	0	22	0	22	
	1	1	10	$q_2=9$	19	
	2	2	0	$2q_2=18$	18*	2
2	0	1	10	0	10	
	1	2	0	$q_2=9$	9*	1
	2	3				

Periode 3 erlaubt
Zustände 0,1,2

unzulässiger Bereich

optimale Werte

Zur Erinnerung:

$g_j(x_j,u_j) = u_j\, q_j$

$q_1=7,\ q_2=9,\ q_3=12,\ q_4=10$

Rückwärtsentwicklung, Periode 1:

x_1	u_1	$x_2=f_1(x_1,u_1)$	$v^*_2(x_2)$	$g_1(x_1,u_1)$	$v_1(x_1,u_1)$	$z^*_1(x_1)$
0	0	-1				
	1	0	28	$q_1=7$	35	
	2	1	18	$2q_1=14$	32*	2

unzulässiger Bereich

optimale Werte

Periode 2 erlaubt nur Zustände 0 und 1

Zur Erinnerung:

$g_j(x_j,u_j) = u_j\, q_j$

$q_1=7,\ q_2=9,\ q_3=12,\ q_4=10$

Vorwärtsentwicklung:

$x^*_1=0 \Rightarrow u^*_1=2 \rightarrow x^*_2=1 \Rightarrow u^*_2=2 \rightarrow x^*_3=2 \Rightarrow u^*_3=0 \rightarrow$

$x^*_4=1 \Rightarrow u^*_4=0 \rightarrow x^*_5=0$

$\Rightarrow$ optimale Politik: (2,2,0,0) und optimale Zustandsfolge: (0,1,2,1,0)

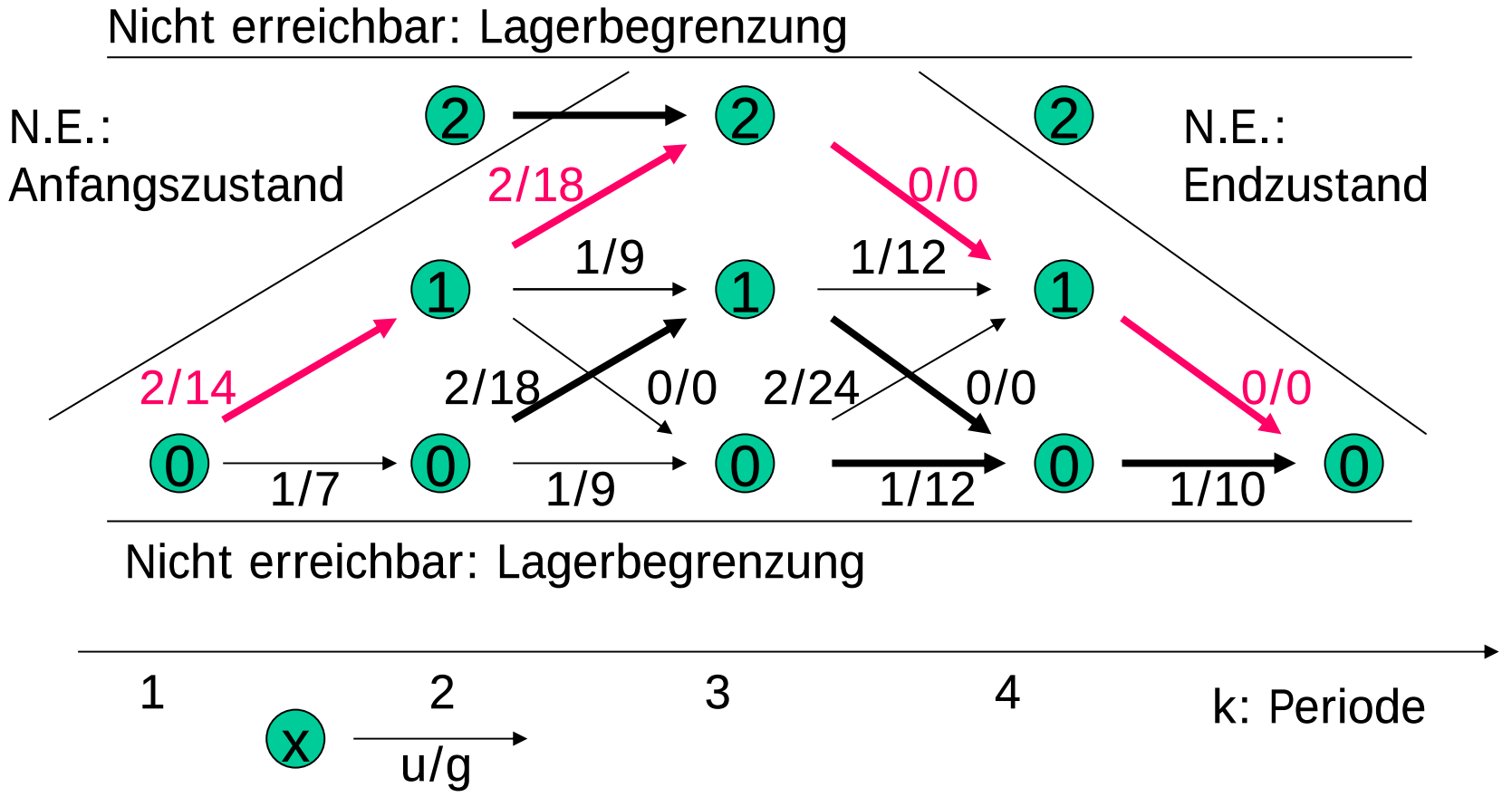
$\Rightarrow$ optimales Ergebnis $v^*_1(0) = 32$

Veranschaulichung der algorithmischen Lösung

Damit optimale Politik bekannt

optimal für $P_j(x_j)$, auf optimaler Entscheidungsfolge

optimal für $P_j(x_j)$, nicht auf optimaler Entscheidungsfolge



Beispiel 2: Spielproblem Verkaufsplanung

Annahmen des Beispiels praktisch nicht voll motivierbar

Planung der Verkaufsmengen

- für 1 Typ von Gut
  - dessen Verfügbarkeitsmenge kontinuierlich anwächst
  - dessen Verkauf i. w. von erzielbaren Erlösen bestimmt ist
- Planung / Verfolgung des Verkaufssystems in Perioden für n = 3 Perioden
  - Gut sei kontinuierlich teilbar
    (kein Stückgut, sondern z.B. Öl, Gas)
    => Verfügbarkeits-, Verkaufsmengen kontinuierlich
- Verkaufsmenge je Periode $u_j \in \mathbb{R}_+$ ME in Periode j
  - Verkauf erfolgt unmittelbar nach Beginn Periode

- Verfügbarkeitsmenge wächst (irgendwie) im Verlauf der Periode mit Faktor a (hier: a=2) mit Verfügbarkeitsmenge $x_j \in \mathbb{R}_+$ ME zu Beginn Periode j => offensichtlich $0 \le u_j \le x_j$
- Zustandsfunktionen (x_{j+1}=) $f_j(x_j,u_j) = a\,(x_j-u_j)$ für j=1,2,3
  Anfangsbestand sei $x_1=1$
- erzielbare Erlöse $g_j(x_j,u_j)$ GE in Periode j
  - wachsen unterlinear mit Verkaufsmenge,
  - aber in jeder Periode identisch
  - gemäß $g_j(x_j,u_j) = \sqrt{u_j} \quad (j=1,2,3)$
- Gesucht: optimale Politik
  - besteht aus Entscheidungen bzgl Verkaufsmengen u_j ME in Periode j
  - führt zu Beständen x_j ME zu Beginn Periode j
  - wahrt die Nebenbedingungen
  - erreicht maximale Erlöse $\sqrt{u_1}+\sqrt{u_2}+\sqrt{u_3}$

Formalisierung.

- Steuerbereiche $U_j(x_j) = [0,x_j]$ für j=1,2,3
- Zustandsbereiche $X_1=\{1\}$, $X_j=\mathbb{R}_+$ für j=2,3,4
- Zustands-(Transformations-)Funktionen
  $(x_{j+1}=)\quad f_j(x_j,u_j) = a\cdot(x_j-u_j)$ für j=1,2,3
- Kostenfunktionen $g_j(x_j,u_j) = \sqrt{u_j} \quad (j = 1,2,3)$
- Durchführung Bellmansche Funktionalgleichungsmethode angepasst an „Maximierung“ und Problem

Im Gegensatz zu den bisherigen Beispielen kontinuierlicher Zustandsraum und Entscheidungsraum!

$$v_j^*(x_j) = \max_{u_j \in U_j(x_j)} \left\{ g_j(x_j,u_j) + v_{j+1}^*[f(x_j,u_j)] \right\}$$
$$= \max_{0 \le u_j \le x_j} \left\{ \sqrt{u_j} + v_{j+1}^*[2(x_j - u_j)] \right\}$$

Einige Vorüberlegungen:

$w(u) := \sqrt{u} + \sqrt{c(x-u)}$ mit $c > 1$ und $x \geq 0$ fest

z(x) := u^+ und u^+ = max [w(u); 0≤u≤x]

w(u) strikt konkav über [0,x]

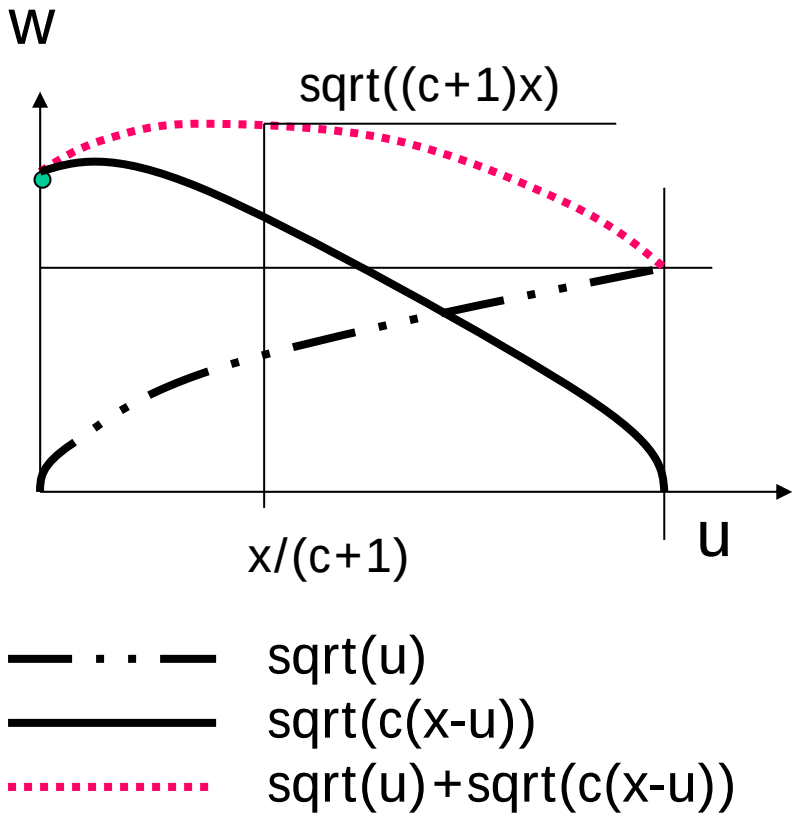
=> Extremstelle in [0,x] ist Maximum

Bestimmung der Extremstelle durch Nullsetzen der ersten Ableitung:

$$w'(u) = \frac{1}{2\sqrt{u}} - \frac{c}{2\sqrt{c(x-u)}} = 0 \Rightarrow$$

$$c\sqrt{u} = \sqrt{c(x-u)} \;\Rightarrow\; u^+ = \frac{x}{c+1}$$

$$w(u^+) = \sqrt{\frac{x}{c+1}} + \sqrt{c\frac{x(c+1)-x}{c+1}} = \sqrt{(c+1)x}$$



Rückwärtsentwicklung:

Periode 3:

- vereinbarungsgemäß ist $v_4^*(x_4) = 0$
- daher $v_3^*(x_3) = \max_{0 \le u_3 \le x_3} (\sqrt{u_3} + 0)$
- $v_3^*(x_3) = \sqrt{x_3},\ z_3^*(x_3) = x_3$ (alles verkaufen)

Periode 2:

- $v_2^*(x_2) = \max_{0 \le u_2 \le x_2} \left(\sqrt{u_2} + \sqrt{2(x_2 - u_2}\right)$
- Maximalstelle für c=2

$$v_2^*(x_2) = \sqrt{3x_2},\ z_2^*(x_2) = x_2/3$$

Periode 1:

- $v_1^*(x_1) = \max_{0 \le u_1 \le x_1} \left(\sqrt{u_1} + \sqrt{6(x_1 - u_1}\right)$
- Maximalstelle für c= 6

$$v_1^*(x_1) = \sqrt{7x_1},\ z_1^*(x_1) = x_1/7$$

Vorwärtsentwicklung:

$x^*_1=1 \Rightarrow u^*_1=1/7 \rightarrow$

$x^*_2=2(x^*_1 - u^*_1)=12/7 \Rightarrow$
$u^*_2= x^*_2 /3 = 4/7 \rightarrow$

$x^*_3=2(x^*_2 - u^*_2)=16/7 \Rightarrow$
$u^*_3= x^*_3 = 16/7 \rightarrow$

$x^*_4=2(x^*_3 - u^*_3)=0$

$\Rightarrow$ optimale Politik
(1/7,4/7, 16/7)

$\Rightarrow$ optimale Zustandsfolge
(1,12/7,16/7,0)

$\Rightarrow$ optimales Ergebnis

$$v_1^*(1) = \sqrt{7} = 2.65$$

Ausblick

Eine Reihe weiterer Aspekte wären noch zu betrachten:

- Aufwandsbetrachtungen
  - wg Rucksackproblem ist exponentieller Aufwand zu erwarten
  - wodurch entsteht der Rechenaufwand ?
    n Stufen, je Stufe X_j Zustände mit Verzweigungsgrad U_j
    Binäres Rucksackproblem: für ganzzahlige Werte, X={0,1,...,A},
    Entscheidung binär, Algorithmus ist pseudopolynomiell: O(nA)
  - Speicherplatzbedarf: n|X| Entscheidungen z*, für 2 Stufen Werte v*
- Verallgemeinerungen
  - bzgl Verknüpfungsoperationen
  - bzgl stochastischer Modelle

Inhalt des nächsten Abschnitts

- Anwendungen
  - Tricks, um Anwendungsprobleme geeignet zu „betrachten“, so dass diese passend formalisierbar sind
  - bekannte / typische Anwendungen, Real-World Examples
  - ...

Detaillierte Informationen z.B. in B. Bertsekas Kurs „Dynamic Programming“ am MIT (http://www.athenasc.com/dpbook.html)

12.4 Stochastische dynamische Programmierung

Bisher untersucht „deterministische“ dynamische Optimierung
System wird

- über endlichen Planungszeitraum aus n Perioden untersucht,
- wo zu Beginn von Periode j Systemzustand x_j herrschte, begrenzt auf Zustandsbereich X_j mit $x_j \in X_j \neq \emptyset$ mit bekanntem Anfangszustand $x_1=x_a$ und $X_1=\{x_1\}$
- wo in Periode j Entscheidung u_j getroffen wurde, begrenzt auf (zustandsabhängigen) Steuerbereich $U_j(x_j)$ und $u_j \in U_j(x_j) \neq \emptyset$
- wo aus Zustand x_j und Entscheidung u_j
  - nächster Zustand x_{j+1} resultierte gemäß $x_{j+1}=f_j(x_j,u_j)$
  - Kosten / Erlöse g_j resultierten gemäß $g_j(x_j,u_j)$
- Einführung probabilistischer / stochastischer Annahmen in Modelle dient
  - (formalisierter) Beherrschung von problemspezifischen Unsicherheiten
  - (aufwandsreduzierender) Einführung von Unsicherheiten in die Problemstellung

Entscheidung u_j auf Basis von Zustand x_j getroffen, führt

- nicht zu eindeutigem Folgezustand x_{j+1}
- sondern zu einem der erlaubten Folgezustände $x_{j+1} \in X_{j+1}$
  gemäß Wahrscheinlichkeitsvert. $pX_j(.|x_j,u_j)$ oder Dichtefkt. $fX_j(.|x_j,u_j)$

<u>Motivierende Betrachtungen</u>

In Bsp 1 könnten

- Auslieferungsmengen statt konstant 1 als schwankend angenommen sein,
- Belieferungskosten statt je Periode j zu q_j €/Stück fest als schwankend angenommen sein

Analog in Bsp 2

- Verfügbarkeitsmengen schwankend (z.B. wg. Ausschuss)
- erzielbare Erlöse nicht präzise sondern in Grenzen schwankend

Unter stochastischen Annahmen

- wird erreichter Folgezustand x_{j+1} erfasst durch Zufallsvariable X_{j+1},
  - bei endlichem / abzählbarem Zustandsbereich X_{j+1} Verteilung charakterisiert (z.B.) durch Menge bedingter Wahrscheinlichkeiten
  - bei kontinuierlichem Zustandsbereich X_{j+1} Verteilung charakterisierbar (z.B.) durch (bedingte) Dichtefunktion
- werden resultierende Kosten/Erlöse
  erfasst durch erweiterte Funktion $g_j(x_j,u_j,x_{j+1})$
  - bzw. als Zufallsvariable G_j mit zugehöriger (bedingter) Verteilung $\{pG_j(g(x_j,u_j))\}$ bzw $fG_j(g(x_j,u_j))$ falls Z_{j+1} diskret bzw kontinuierlich
  - andere Möglichkeit: gemeinsame Verteilung (G_j,X_{j+1})

Bellmansche Funktionalgleichung

Unter diesen Annahmen (Ziel: Bellmansche Fktgl.)

- ist der minimal Zielfunktionsbeitrag der Perioden j,...,n
  von realisiertem Zustand x_j zu Beginn Periode j startend
  ebenfalls als Zufallsvariable $V^*_j(x_j)$ zu charakterisieren, mit ihrer Verteilung
  und ihrem Erwartungswert $E(V^*_j(x_j))$
- setzt sich der Erwartungswert des minimalen Zielfunktionsbeitrags (Perioden j,...,n) von Zustand x_j startend, bei Realisierung eines Folgezustands (additiv) zusammen gemäß $g_j(x_j,u_j,x_{j+1})+E(V^*_{j+1}(x_{j+1}))$
- ergibt sich die Bellmansche Funktionalgleichung analog zur Ableitung der deterministischen Form bei diskretem Zustandsbereich X_{j+1}

$$E(V_j^*(x_j)) = \min_{u_j \in U_j(x_j)} \left(\sum_{x \in X_{j+1}} \left(g_j(x_j, u_j, x) + E(V_{j+1}^*(x)) \right) pX_j(x|x_j, u_j) \right)$$

Allgemeine Form mit kontinuierlichen Wertebereichen:

$$E(V_j^*(x_j)) = \min_{u_j \in U_j(x_j)} \left(\int_{x \in X_{j+1}} \left(g_j(x_j, u_j, x) + E(V_{j+1}^*(x)) \right) f X_j(x|x_j, u_j) dx \right)$$

Unter Einsatz des Erwartungswertes

$$E(g_j(x_j, u_j)) = \int_{x \in X_{j+1}} g_j(x_j, u_j, x) f X_j(x|x_j, u_j) dx$$

wird daraus

$$E(V_j^*(x_j)) = \min_{u_j \in U_j(x_j)} \left(E(g_j(x_j, u_j)) + \int_{x \in X_{j+1}} E(V_{j+1}^*(x)) f X_j(x|x_j, u_j) dx \right) \quad (*)$$

Oft nicht analytisch lösbar $\Rightarrow$

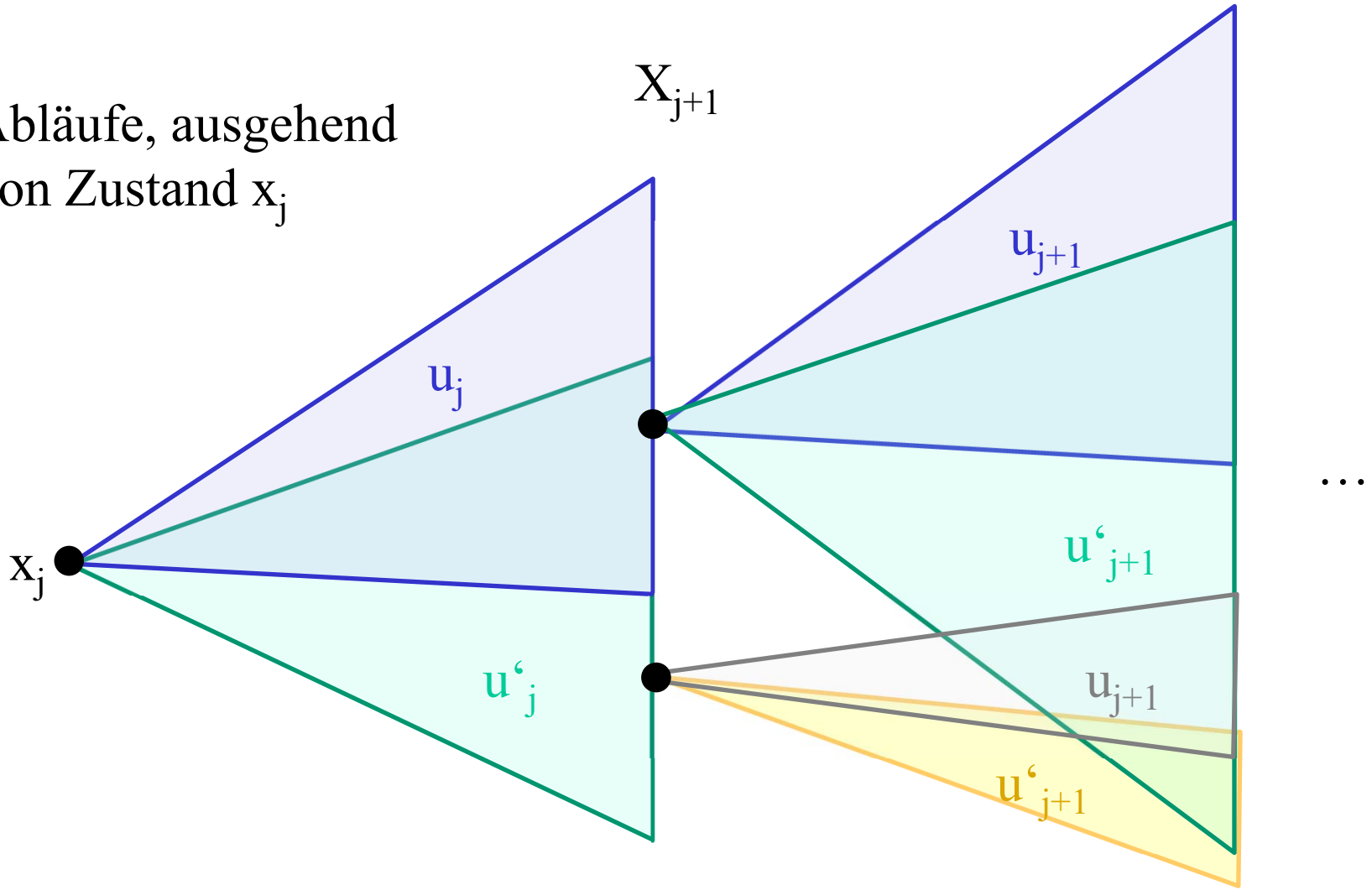
Approximation oder Simulation einsetzen

Analysemöglichkeiten für (*):

1. Ersetzung des zufälligen Nachfolgezustand durch den Erwartungswert
2. Eingeschränkter Blick in die Zukunft
   (engl. limited lookahead)
3. Simulation einzelner Trajektorien
4. Festlegung einer suboptimalen Politik zur Vorwärtsberechnung

Kombinationen aus diesen 4 Ansätzen

Abläufe, ausgehend von Zustand x_j



Zufallsgesteuerte Sichtweise des Nachfolgezustands durch
Zufallsparameter w ∈ W_j und Funktion f(x_j, u_j, w)
($\delta(y \in Y) = 1$ falls $y \in Y$ und 0 sonst)

$$\int_{w \in W_j} \delta\left(f(x_j, u_j, w) \in Y_{j+1}\right) dw = \int_{x \in Y_{j+1}} fX_j(x|x_j, u_j)dx \text{ für alle } Y_{j+1} \subseteq X_{j+1}$$

1. Ersetzung durch den Erwartungswert

$$\bar{x}_{j+1} = \int_{w \in W_j} f(x_j, u_j, w)dw$$

Rechnen mit den Erwartungswerten liefert ein deterministisches Problem

2. Eingeschränkter Blick in die Zukunft (um eine Periode)

Ersetze $E(V^*_{j+1}(x))$ durch

$$\bar{V}_j\left(x_j, u_j\right)=\int_{x\in X_{j+1}} \min_{u_{j+1}\in U_{j+1}(x)}\left(E\left(g_{j+1}(x, u_{j+1})\right) f X_j\left(x\middle|x_j, u_j\right) dx\right.$$

Lösung vernachlässigt zukünftige Ereignisse nach j+1

Lösung des Integrals i.d.R. nur numerisch/simulativ

Prinzipiell auch auf mehr als einen Schritt in die Zukunft ausdehnbar

Beispiel 2 Schritte

$$\bar{V}_j\left(x_j, u_j\right)=\int_{x\in X_{j+1}} \min_{u_{j+1}\in U_{j+1}(x)}\left(E\left(g_{j+1}(x, u_{j+1})\right) + \int_{y\in X_{j+2}} \min_{u_{j+2}\in U_{j+2}(y)} E\left(g_{j+2}\left(y, x_{j+2}\right)\right) f X_{j+1}(y|x, u_{j+1} dy)) f X_j\left(x\middle|x_j, u_j\right) dx\right.$$

Sehr aufwändig, kaum in geschlossener Form auswertbar

3. Simulation

im Prinzip Monte-Carlo Methode zur Auswertung der Integrale

sei $w_j^{(k)}$ Zufallswert in Stufe j und Replikation k,

$f\left(x_j, u_j, w_j^{(k)}\right)$ Folgezustand (deterministisch bei gegebenem $w_j^{(k)}$)

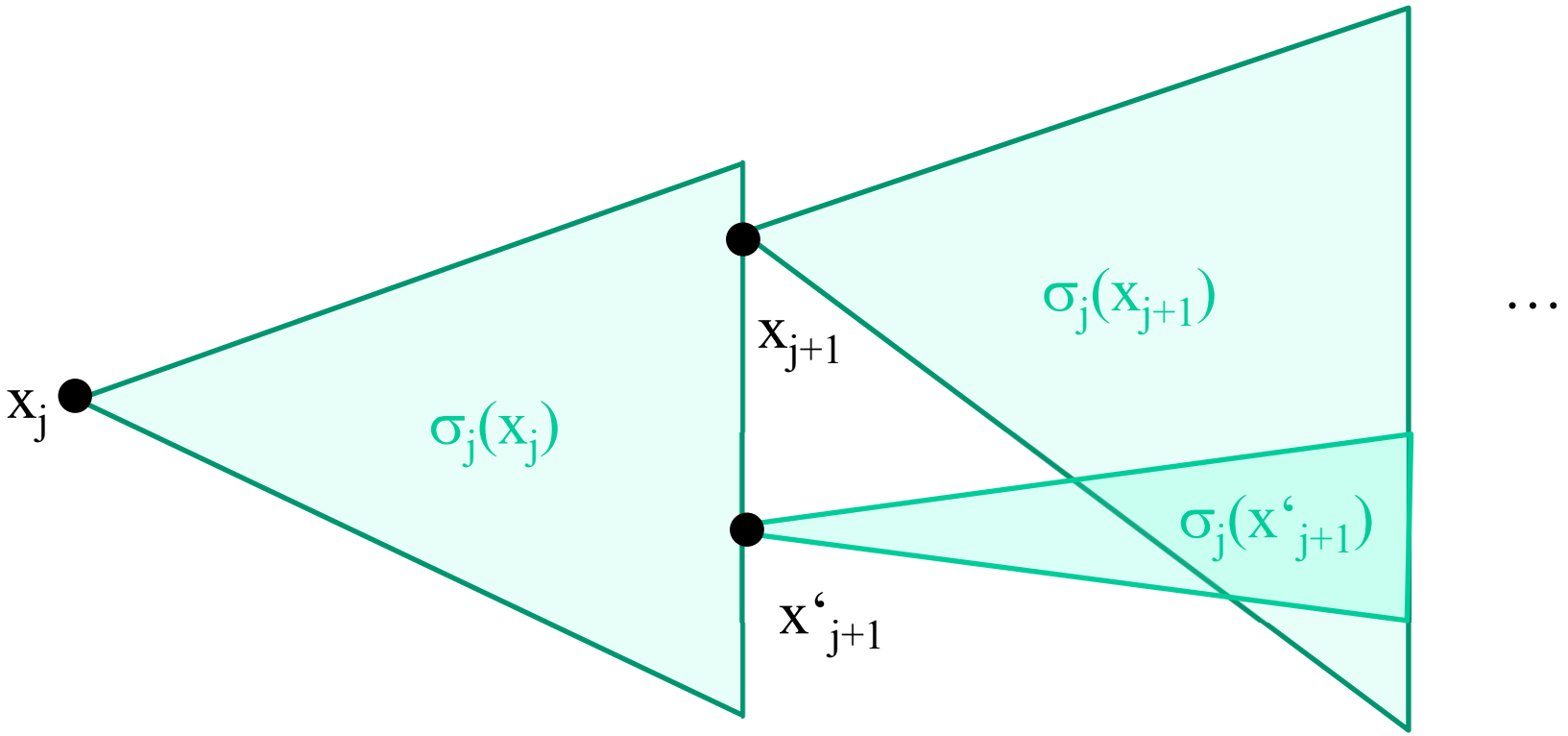
$$\bar{V}_j(x_j, u_j) \approx \frac{1}{K} \sum_{k=1}^{K} \min_{u_{j+1} \in f(x_j, u_j, w_j^{(k)})} \left(E(g_{j+1}(f(x_j, u_j, w_j^{(k)})), u_{j+1} \right).$$

Erweiterung auf einen weitergehenden Blick in die Zukunft möglich

Dann K_j Replikationen mit den Schritten

1. Bestimmung von x_j mittels $w_j^{(k)}$
2. Ausgehend von x_j führe K_{j+1} Replikationen zur Bestimmung von x_{j+1} mittels $w_{j+1}^{(k,l)}$ durch
3. Wähle bzgl. der in 2. durchgeführten Replikationen bestes u_j in Replikation k

Um den Blick in die Zukunft zu verlängern, wird i.d.R. eine plausible Politik σ vorgegeben und nicht in jedem Schritt optimiert!



Bisheriger Ansatz minimiert den Erwartungswert,

Andere Zielgröße: Minimierung des maximalen Wertes

Zugehörige Darstellung des Optimierungsproblems

$$E(V_j^*(x_j)) = \min_{u_j \in U_j(x_j)} \left(E(g_j(x_j, u_j)) + \max_{w \in W_j} E(V_{j+1}^*(f(x_j, u_j, w)) \right)$$

Analysemethoden im Prinzip wie vorher

Deutlich schwieriger sind Probleme, bei denen nicht das Maximum, sondern ein Wert der mit vorgegebener Wahrscheinlichkeit unterschritten werden muss, minimiert wird

Oft sollen die Kosten über einen unendlichen Zeithorizont minimiert werden (siehe stationäre Analyse bei der Simulation)

$\Rightarrow$ Summierte Kosten konvergieren i.d.R, gegen ∞

Mögliche Sichtweise

- Kosten in der Gegenwart sind schlechter als Kosten in der Zukunft (abgezinste Werte)

$$E(V_j^*(x_j)) = \min_{u_j \in U_j(x_j)} \left(E(g_j(x_j, u_j)) + \alpha \int_{x \in X_{j+1}} E(V_{j+1}^*(x)) f X_j(x|x_j, u_j) dx \right)$$

mit $\alpha<1$

Ziel ist oft die Bestimmung von $E(V^*(x)) = \lim_{j \to \infty} E(V_j^*(x))$

und der zugehörigen Politik

Die Berechnung der stationären Politik ist oft „einfacher“ als die Bestimmung der schrittabhängigen Politik!

Annahmen

- es gelte für den Zustandsraum $X=X_j = X_{j+k}$ für ein $j \geq 0$ und alle $k \geq 0$
- U(x) sei so gewählt, dass fX(.|x,u) eine messbare Funktion für alle $x \in X$ und $u \in U(x)$ ist

Dann gilt

$$E(V^*(x)) = \min_{u \in U(x)} \left(E(g(x,u)) + \alpha \int_{y \in X} E(V^*(y)) fX(y|x,u) \right)$$

d.h. die optimale Politik hängt nur vom Zustand ab
(stationäre Politik)

12.5 Markovsche Entscheidungsprozesse über endlichen Horizonten

Zustandsraum $X = \{1,...,m\}$ und

Steuerbereich $U(i) = \{u_{i1},...,u_{is_i}\}$ $(i = 1,...,m)$

Unabhängig von der Periode!

Verhaltensbeschreibung:

Sei x_j=i Zustand zu Beginn von Periode j

- System geht zu Beginn von Periode j+1 mit Wahrscheinlichkeit $p_j(i,u,k)$ in Zustand x_{j+1}=k bei Entscheidung $u \in U(i)$ über
- es gilt für die Übergangswahrscheinlichkeiten $p_j(i,u,k)$:
  $\sum_{k=1}^{m} p_j(i,u,k) = 1$ für alle $i \in X$ und $u \in U(i)$
- Politik wird definiert durch Vektoren $\sigma_1,\ldots,\sigma_n$ mit $\sigma_j(i) \in U(i)$

Für eine vorgegeben Politik beschreibt der Markovsche Entscheidungsprozess (ohne Betrachtung der Kosten) einen Markov Prozess

Einschub Markov Prozesse:

Y_j sei eine Sequenz von Zufallsvariablen, die Werte über einem abzählbaren (hier endlichem) Zustandsraum X={1,…,m} annehmen

Y_j (j=1, 2, ..) ist ein Markov Prozess, genau dann wenn

$$P\left[Y_j = i_j | Y_1 = i_1, \ldots, Y_{j-1} = i_{j-1}\right] = P\left[Y_j = i_j | Y_{j-1} = i_{j-1}\right]$$

Wir schreiben $p_j(i,k) = P\left[Y_{j+1} = k | Y_j = i\right]$

Im hier behandelten zeitdiskreten Fall spricht man auch von Markov Ketten

Zusammenfassung der Übergangswahrscheinlichkeiten in

m x m Matrix P_j (nicht negativ, Zeilensumme 1 d.h. stochastische Matrix)

- Falls die Übergangswahrscheinlichkeiten von der Periode abhängen ⇒ inhomogener Markov Prozess
- Falls die Übergangswahrscheinlichkeiten nicht von der Periode abhängen ⇒ homogener Markov Prozess
  (Index j weglassen)

Beschreibung des Verhaltens mittels Vektoren und Matrizen:

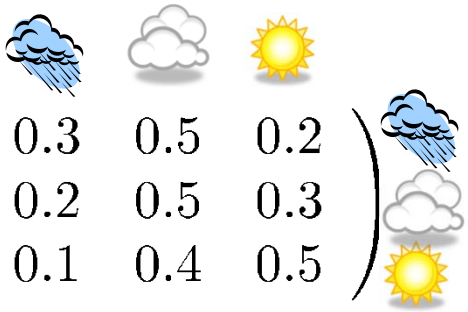
π_j Zustandsvektor in Periode j,

$\pi_j(i)$ Zustandswahrscheinlichkeit Zustand i in Periode j

Es gilt:

$$\pi_{j+1} = \pi_1 \prod_{k=1}^{j} P_k$$

Ein kleines Beispiel (Wetter in Dortmund)
Zustandsraum S = {1, 2, 3}
x = 1: Regen
x = 2: Bedeckt
x = 3: Sonnig



$$\mathbf{P} = \begin{pmatrix} 0.3 & 0.5 & 0.2 \\ 0.2 & 0.5 & 0.3 \\ 0.1 & 0.4 & 0.5 \end{pmatrix}$$

Beispiele: $p_{11} = 0.3$ wenn es heute regnet, regnet es mit Wahrscheinlichkeit 0.3 auch morgen

$p_{31} = 0.1$ wenn es heute sonnig ist, regnet es morgen mit Wahrscheinlichkeit 0.1

Beispiel (Wetter in Dortmund in 2 Tagen)

$$\mathbf{P}^2 = \begin{pmatrix} 0.21 & 0.48 & 0.31 \\ 0.19 & 0.47 & 0.34 \\ 0.16 & 0.45 & 0.39 \end{pmatrix}$$

Bsp. Wenn heute ein sonniger Tag ist, so ist mit Wahrscheinlichkeit 0.39 übermorgen ein sonniger Tag!

Zustandsklassifizierung für homogene Markov Prozesse
(mit endlichem Zustandsraum):

- Zustände i und k kommunizieren gdw.
  a,b ∈ℕ existieren, so dass $P^a(i,k)>0$ und $P^b(k,i)>0$
- $X^c \subseteq X$ Klasse von Zuständen, so dass
  alle i,k ∈ X^c kommunizieren und
  für alle i ∈ X^c und k ∉ X^c gilt P(i,k) = 0
  X^c ist eine rekurrente Klasse von Zuständen
- Der Zustandsraum eines Markov Prozesses zerfällt in eine Menge rekurrenter Klassen und die restlichen Zustände, die in einer transienten Klasse zusammengefasst werden
- Falls der Zustandsraum nur eine rekurrente Klasse beinhaltet, so existiert die stationäre Verteilung als eindeutige Lösung von

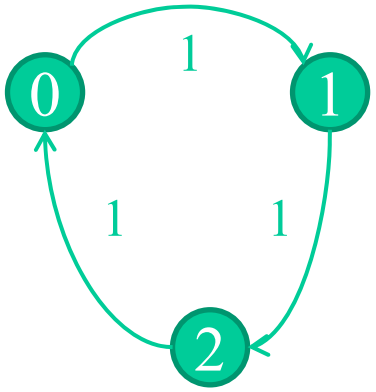
$$\pi P = \pi \text{ und } \sum_{i=1}^{m} \pi(i) = 1$$

Sei $\mathcal{P}_i = \{j > 0 | P^j(i,i) > 0\}$ und sei d der größte gemeinsame Teiler der Elemente

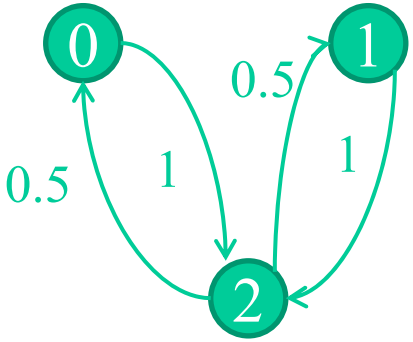
Falls d>1, so ist Zustand i periodisch, ansonsten aperiodisch

Der Markov Prozess heißt aperiodisch, falls alle Zustände aperiodisch sind

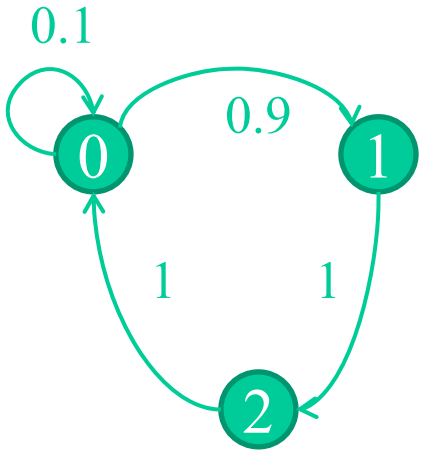
Alle Zustände einer rekurrenten Klasse haben die selbe Periode



Periodisch mit Periode d=3



Periodisch mit Periode d=2



Aperiodisch

Falls der Markov Prozess aperiodisch ist (d.h. alle Zustände sind aperiodisch), so existiert $\lim_{n\to\infty} P^n$

Sei $P^* = \lim_{n\to\infty} \frac{1}{n} \sum_{j=1}^{n} P^n$ dann gilt:

- Falls k ein transienter Zustand ist, so ist für alle $i \in X$: $P^*(i,k) = P^*(k,i) = 0$
- Falls i und k zu unterschiedlichen rekurrenten Klassen gehören, so ist $P^*(i,k) = P^*(k,i) = 0$
- Falls i und k zur selben rekurrenten Klasse gehören, so ist für alle h$\in X$: $P^*(i,h) = P^*(k,h)$
- Falls der Zustandsraum X nur eine rekurrente Klasse von Zuständen beinhaltet, so sind alle Zeilen von P^* gleich π (der stationären Verteilung, d.h. $\pi P = \pi$ und $\pi 1 = 1$)

Ende Einschub

Für eine vorgegebene Politik $\sigma_1,\ldots,\sigma_n$ beschreibt ein Markovscher Entscheidungsprozess einen inhomogenen Markov Prozess

Zusätzlich werden zur Optimierung Kosten definiert

$g_j(i,u,k) \geq 0$ sind die Kosten, die in Periode j auftreten, wenn im Zustand i Entscheidung u getroffen wird und der Nachfolgezustand j ist

Erwartete Kosten der Entscheidung u in Zustand i:

$$E(g_j(i,u)) = \sum_{k=1}^{m} p_j(i,u,k) g_j(i,u,k)$$

Bellmansche Funktionsgleichung:

$$E(V_j(i)) = \min_{u \in U(i)} \left(E(g_j(i,u)) + \sum_{k=1}^{m} p_j(i,u,k) E(V_{j+1}(k)) \right)$$

Sei $u^*_j(i)$ eine Entscheidung u, für die das Minimum $E(V^*_j(i))$ angenommen wird
(also eine optimale Entscheidung im Zustand i für Periode j)

Bestimmung von $u^*_j(i)$ und $E(V^*_j(i))$:

- Auswertung der Funktionsgleichung rückwärts von j = n bis 1 ausgehend von $E(V^*_{n+1}(i)) = 0$ für i = 1,...,m
- Anschließende Vorwärtsberechnung von j = 1 bis n ermittelt die optimale Politik und die dadurch auftretenden Zustandswahrscheinlichkeiten (keine eindeutige Zustandsfolge!)

Vorgehen im Prinzip analog zum deterministischen Fall

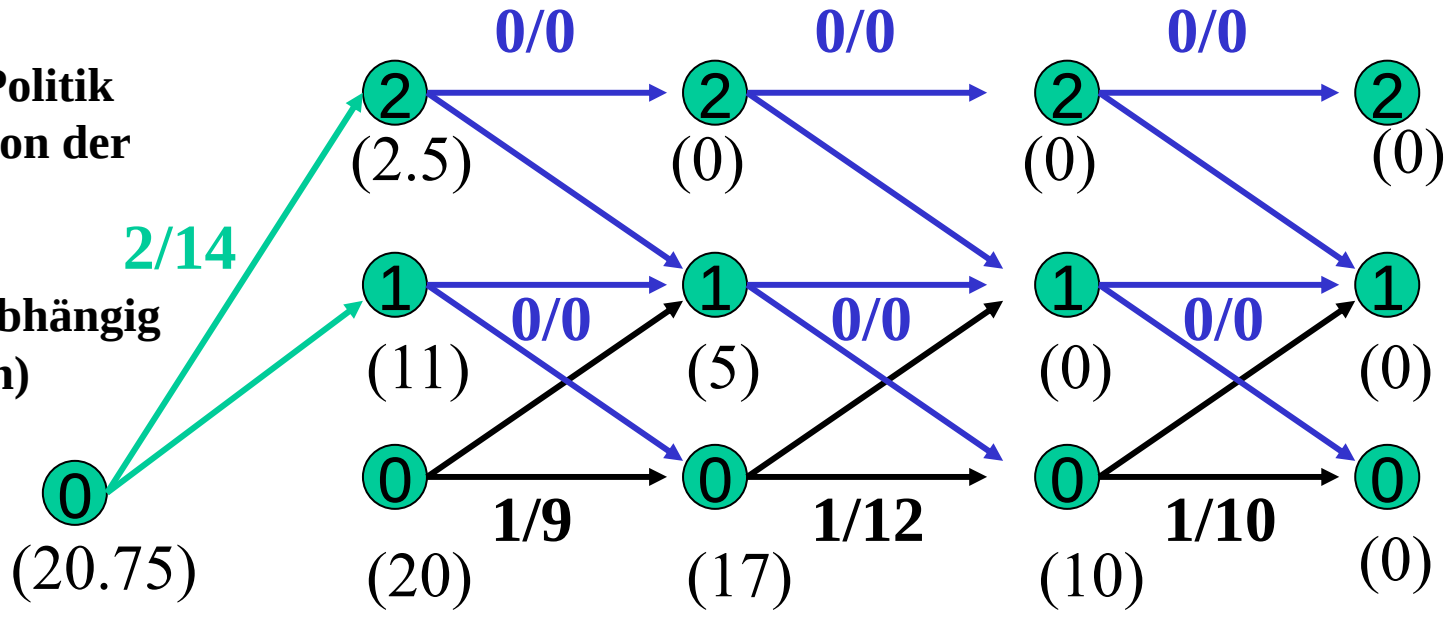
Erweiterung des Beispiels von Folie 27-32

Zus. Annahme mit Wahrscheinlichkeit 0.5 wird ein Teil in der Periode ausgeliefert und mit Wahrscheinlichkeit 0.5 wird kein Teil ausgeliefert

Endzustand 0 am Ende der vierten Periode kann damit nicht gefordert werden!

Folgende Graphik zeigt nur die optimalen Entscheidungen in einem Zustand in der Form u/g

Optimale Politik abhängig von der Nachfrage (d.h. situationsabhängig entscheiden)



Man unterscheidet folgende Politiken:

1. Stationär: Politik hängt nur vom Zustand ab
2. Deterministisch: in einer Situation wird eine Entscheidung getroffen
3. Randomisiert: die Entscheidung wird mit Hilfe einer Wahrscheinlichkeitsverteilung über die möglichen $u \in U_i$ getroffen
4. Markovsch: die Entscheidung hängt nur vom aktuellen Zustand (inkl. der Restzeit T-t) ab
5. Allgemein (*history dependent*): die Entscheidungen können auch von in der Vergangenheit getroffenen Entscheidungen abhängen

Wir betrachten nur Markovsche deterministische Politiken
Wann gehört die optimale Politik zu dieser Klasse?

Schreibweise für Markovsche Entscheidungsprozesse

$\sigma = \{\sigma_0, \sigma_1, \sigma_2, \ldots\}$ Politik, wobei $\sigma_j(i) \in U_j$

$$\mathbf{P}_j^{\sigma_j} = \begin{pmatrix} p_j(1,\sigma_j(1),1) & \cdots & p_j(1,\sigma_j(1),m) \\ \vdots & \vdots & \vdots \\ p_j(m,\sigma_j(m),1) & \cdots & p_j(m,\sigma_j(m),m) \end{pmatrix} \quad \mathbf{g}_j^{\sigma_j} = \begin{pmatrix} E(g_j(1,\sigma_j(1))) \\ \vdots \\ E(g(m,\sigma_j(m))) \end{pmatrix}$$

wobei σ_j eine Vektor der Länge m mit $\sigma_j(i) \in U(i)$ ist

Es gilt dann bei Wahl von Politik σ_j in Schritt j

$$\mathbf{v}_j = \mathbf{g}_j^{\sigma_j} + \mathbf{P}_j^{\sigma_j} \mathbf{v}_{j+1}$$

Berechnung durch Vektor-Matrix Multiplikation

Berechnung der optimalen Politik

1. Initialisiere $\boldsymbol{v}_K, k = K$
2. $\boldsymbol{v}_{k-1} = \min_{\mu}(\boldsymbol{g}^{\mu} + \boldsymbol{P}^{\mu}\boldsymbol{v}_k)$;
3. $\sigma_{k-1} = \arg\min_{\mu}(\boldsymbol{g}^{\mu} + \boldsymbol{P}^{\mu}\boldsymbol{v}_k)$;
4. Falls $k > 1$ setze $k = k - 1$ und fahre bei 2. fort;
5. $\sigma_1, \dots, \sigma_K$ enthält die optimale Politik
   $\boldsymbol{v}_1, \dots, \boldsymbol{v}_K$ die optimalen Werte

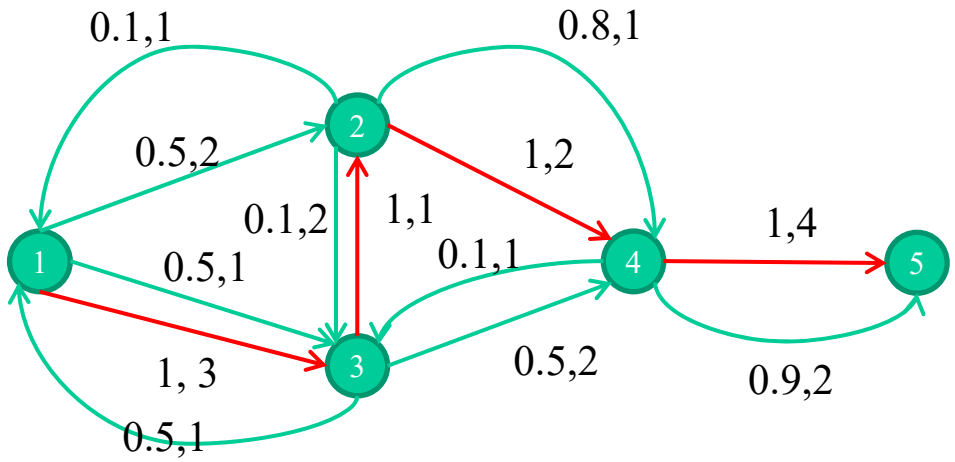
Einfache Realisierung, aber u.U. aufwändig durch große Zustandszahl

Ermittelte Politik ist optimal (da U_i endlich, g nur vom Zustand abhängig)

Anwendungsbeispiel: Stochastisches kürzeste Wege-Problem:

Entscheidung für rote oder grüne Kanten

Ziel: Zustand 5 mit möglichst geringen Kosten in höchsten K Schritten zu erreichen



Schreibweise (p,g)
p Wahrscheinlichkeit
g Kosten

Es gilt hier: Für k→∞ konvergiert die Wahrscheinlichkeit im Zustand 5 zu sein gegen 1 (unabhängig davon, ob rot oder grün gewählt wurde).

12.6 Markovsche Entscheidungsprozesse über unendlichen Horizonten

Bisherige Betrachtung fokussierte auf einen endlichen Planungshorizont und betrachtete den akkumulierten Gewinn oder die akkumulierten Kosten

Oft sollen Gewinne/Kosten über einen unendlichen Planungshorizont untersucht werden, da Systeme im stationären Zustand analysiert und gesteuert werden sollen

Das bisherige Vorgehen offensichtlich nicht anwendbar, da

- Gewinne/Kosten unendlich wären
- unendliche Zustandssequenzen analysiert werden müssten

Alternative Interpretation von Kosten/Gewinn:

- Erwartungswert pro Schritt
- Gewinne werden durch Diskontierungsfaktor $\alpha \in (0,1)$ geschmälert
  d.h. Gewinn x in k Perioden ist jetzt nur $\alpha^k \cdot x$ wert

Analysemethoden:

- Werteiteration
- Lineare Programmierung
- Politikiteration

Bestimmung des Erwartungswerts pro Schritt:

- $v^*_j(i)$ sind die minimalen Kosten im Zustand i nach j Schritten
- $c^+_j(i) = v^*_j(i) / j$ minimale Kosten pro Schritt
- $\sigma^+_j = (\sigma^+_j(1), \ldots, \sigma^+_j(m))$ ist eine Politik, die im *j*ten Schritt angewendet wird und die minimalen Kosten $v^*_j(i)$ erreicht

---

Annahmen:

Die Kosten und Übergangswahrscheinlichkeiten hängen nicht von der Periode ab

Es existiert ein Zustand $i \in X$ und eine ganze Zahl $n>0$, so dass, ausgehend von jedem Startzustand $k \in X$ und unter jeder stationären Politik σ, i innerhalb von n Schritten mit positiver Wahrscheinlichkeit erreicht wird.

Dann existiert eine optimale stationäre Politik

Sei σ=(σ(1),…,σ(m)) eine stationäre deterministische Politik (Politik-/Entscheidungsvektor) dann hat unter obiger Annahme das folgende Gleichungssystem eine eindeutige Lösung:

$$\pi_\sigma(i) = \sum_{k=1}^{m} \pi_\sigma(k) P^\sigma(k,i) \quad \text{und} \quad \sum_{k=1}^{m} \pi_\sigma(k) = 1$$

π_σ ist die stationäre Verteilung des Markov-Prozesses unter Entscheidungsvektor σ

D.h. σ(i)∈U(i)

Es gilt ferner:

$$\pi_\sigma(i) = \lim_{j\to\infty} \pi_\sigma^j(i) \text{ mit } \pi_\sigma^j(i) = \sum_{k=1}^{m} \pi_\sigma^{j-1}(k) P^\sigma(k,i) \ , \ \pi_\sigma^0 \geq 0, \ \sum_{k=1}^{m} \pi_\sigma^0(k) = 1$$

falls der resultierende Markov Prozess aperiodisch ist!

Grundlagen zur Bestimmung optimaler Politiken:

Man kann zeigen (ohne dass wir es tun werden), dass

- $\lim_{j\to\infty} c^+_j(i) = c^+(i) = E(C^+)$ und $E(C^+)$ die minimalen erwarteten Kosten des Markovschen Entscheidungsproblems mit unendlichem Planungshorizont sind
- eine stationäre optimale Politik $\boldsymbol{\sigma}^+ = (\sigma^+(1),...,\sigma^+(m))$ existiert, so dass bei Wahl von Entscheidung $\sigma^+(i)$ in Zustand i bei einem unendlichen Planungszeitraum die erwarteten Kosten $E(C^+)$ auftreten
- $E(C^+)$ und $\boldsymbol{\sigma}^+$ die folgenden Gleichungen erfüllen

$$\pi_{\sigma^+}(i) = \sum_{k=1}^{m} \pi_{\sigma^+}(k) P^{\sigma^+}(k,i) \quad \text{und} \quad \sum_{k=1}^{m} \pi_{\sigma^+}(k) = 1$$

$$E(C^+) = \sum_{i=1}^{m} \pi_{\sigma^+}(i) E(g(i,\sigma^+(i)) \text{ mit } E(g(i,\sigma^+(i)) = \sum_{k=1}^{m} P^{\sigma^+}(i,k) g(i,\sigma^+(i),k)$$

Werteiteration

Idee: Nutze die Rückwärtsphase der dynamischen Programmierung und wähle jeweils die aktuell beste Politik

$\Rightarrow$ Iterationsverfahren mit Iterationsvorschrift

$$v_j(i) = \min_{\sigma(i) \in U(i)} \left(E(g(i, \sigma_i)) + \sum_{k=1}^{m} p(i, \sigma(i), k) v_{j-1}(k) \right)$$

für beliebige initiale Kosten v_0 konvergiert das Verfahren, also
$\lim_{j \to \infty} v_j(i)/j = E(C^+)$ und die dabei berechnete Politik konvergiert gegen σ^+

Abbruchbedingung für die Iteration:

1. Politik ändert sich nicht mehr
2. $\left| \frac{j \cdot v_{j+1}(i)}{\max(\eta, (j+1) \cdot v_j(i))} - 1 \right| < \epsilon$ für alle $i \in X$ und ein $\varepsilon > 0$

Nachteil des bisherigen Vorgehens: I.d.R. gilt $\lim_{j\to\infty} v_j(i) = \infty$

Relative Werteiteration

Wähle festen Bezugszustand s∈X und iteriere

$$h_j(i) = \min_{\sigma_i \in U(i)} \left(E(g(i,\sigma_i)) + \sum_{k=1}^{m} p(i,\sigma_i,k) h_{j-1}(k) \right) - \min_{\sigma_s \in U(s)} \left(E(g(s,\sigma_s)) + \sum_{k=1}^{m} p(s,\sigma_s,k) h_{j-1}(k) \right)$$

mit $h_0(i) = v_0(i) - v_0(s)$ (es gilt dann $h_j(i) = v_j(i) - v_j(s)$)

Abbruchbedingung für die Iteration:

1. Politik ändert sich nicht mehr
2. $\left| \frac{h_{j+1}(i)}{\max(\eta, h_j(i))} - 1 \right| < \epsilon$ für alle i∈X und ein ε>0

$$E(C^+) = \min_{\sigma_s \in U(s)} \left(E(g(s,\sigma_s)) + \sum_{k=1}^{m} p(s,\sigma_s,k) h(k) \right)$$

Darstellung als lineares Programm:

Sei y_{iu} ($i \in \{1,...,m\}$, $u \in U(i)$) die Wahrscheinlichkeit im Zustand i zu sein und Entscheidung u zu treffen

Es gilt (lineare Nebenbedingungen):

$$\sum_{i=1}^{m} \pi(i) = \sum_{i=1}^{m} \sum_{u \in U(i)} y_{iu} = 1.0$$

und

$$\sum_{u \in U(i)} y_{iu} = \sum_{k=1}^{m} \sum_{u' \in U(i)} y_{ku'} p(k,u',i)$$

Außerdem muss $y_{i\sigma} \geq 0$ gelten

Zielfunktion: $\min \left(\sum_{i=1}^{m} \sum_{u \in U(i)} y_{iu} E(g(i,u)) \right)$

Mittels Simplex-Algorithmus lösbar

Variablenzahl: $\Sigma_{i=1,..m} s_i$

Aus dem bisher Gesagten folgt, dass für die optimale Lösung

$y_{iu} > 0$ für genau ein u und für alle anderen u‘ $\in U(i) \setminus \{u\}$ $y_{iu'}=0$

Lösung mittels Politikiteration:

Für eine stationäre Politik σ = (σ(1),..,σ(m)) gilt

$$G(\sigma) + v(i) = E(g(i, \sigma(i))) + \sum_{k=1}^{m} P^{\sigma}(i,k) v(k) \ (i = 1, \ldots, m)$$

wobei $G(\sigma) = \sum_{k=1}^{m} E(g(k, \sigma(k))) \pi_{\sigma}(k)$ und $\pi_{\sigma}(k) = \sum_{i=1}^{m} P^{\sigma}(i,k) \pi_{\sigma}(i)$

Erläuterung des Zusammenhangs für eine feste Politik σ

Für große n gilt $v_n(i) \approx n \cdot G(\sigma) + v(i)$

und damit auch $v_n(i) - v_n(j) \approx v(i) - v(j)$

Eingesetzt in die rekursive Beziehung ergibt sich
(Vorsicht, es wird rückwärts von n aus gerechnet):

$$\begin{aligned}
v_n(i) &= E((g(i, \sigma(i))) + \sum_{k=1}^{m} P^{\sigma}(i,k) v_{n-1}(k) & \Leftrightarrow \\
nG(\sigma) + v(i) &= E(g(i, \sigma(i))) + \sum_{k=1}^{m} P^{\sigma}(i,k) \left((n-1)G(\sigma) + v(k)\right) & \Leftrightarrow \\
G(\sigma) + v(i) &= E(g(i, \sigma(i))) + \sum_{k=1}^{m} P^{\sigma}(i,k) v(k)
\end{aligned}$$

„Ausprobieren“ aller möglichen Politiken führt zu einem Algorithmus mit Aufwand Πs_i (inakzeptabel)

Besser schrittweise Verbesserung der Politik

1. starte mit einer Politik $\boldsymbol{\sigma} = (\sigma(1),...,\sigma(m))$ wähle eine gute Näherungspolitik oder wähle $\sigma(i)$ so dass $E(g(i, \sigma(i))$ minimal
2. berechne die zugehörigen Werte $\mathbf{v} = (v(1),...,v(m))$
   v ist die Lösung des linearen Gleichungssystems
   $$G(\sigma) = E(g(i,\sigma(i))) + \left(\sum_{k=1}^{m} P^{\sigma}(i,k)v(k)\right) - v(i)$$
   wobei $v(m) = 0$ gesetzt wird (sonst keine eindeutige Lsg.)
3. bestimme verbesserte Politik $\boldsymbol{\sigma}$‘ aus der folgenden Minimierung
   $$\min_{\sigma'(i)\in U(i)} \left(E(g(i,\sigma'(i))) + \sum_{k=1}^{m} p(i,\sigma'(i),k)v(k)\right) \qquad \text{für } i = 1,...,m$$
4. Falls $\boldsymbol{\sigma}$‘=$\boldsymbol{\sigma}$ ist eine optimale Politik erreicht, sonst setze $\boldsymbol{\sigma}$=$\boldsymbol{\sigma}$‘ und fahre bei 2. fort (es gilt dann $G(\boldsymbol{\sigma}‘) < G(\boldsymbol{\sigma})$)

Analyse der Kosten bei Multiplikation mit Diskontierungsfaktor $\alpha < 1$ (spätere Kosten sind positiv)

Bellmansche Funktionsgleichungen:

$$v_j^+(i) = \min_{u \in U(i)} \left(E(g(i,u)) + \alpha \sum_{k=1}^{m} p(i,u,k) v_{j+1}(k) \right)$$

Ziel:

Ermittlung einer optimalen Politik, so dass bei einem unendlichen Planungshorizont $\lim_{n \to \infty} v^+_n(i)$ minimiert wird
(durch Diskontierung bleiben Kosten endlich!)

Mögliche Lösungsmethoden:

- Wertiteration
- Politikiteration

Grundlagen:

Man kann zeigen (ohne dass wir es tun werden), dass

- $\lim_{j\to\infty} v^+_j(i) = v^+(i)$ und $v^+(i)$ die minimalen diskontierten erwarteten Kosten des Markovschen Entscheidungsproblems mit unendlichem Planungshorizont sind
- eine stationäre optimale Politik $\boldsymbol{\sigma}^+ = (\sigma^+(1),...,\sigma^+(m))$ existiert, so dass bei Wahl von Entscheidung $\sigma^+(i)$ in Zustand i bei einem unendlichen Planungszeitraum die erwarteten Kosten
  $v^+ = (v^+_1,...,v^+_m)$ auftreten
- v^+ und $\boldsymbol{\sigma}^+$ erfüllen die Gleichungen

$$v^+(i) = E(g(i,\sigma^+(i)) + \alpha \cdot \sum_{k=1}^{m} P^{\sigma^+}(i,k)v^+(k)$$

Werteiteration (schrittweise Approximation der optimalen Kosten):

1. Initialisiere $v^+_0(i) = 0$ für alle i = 1,...,m und setze j=1 ;
2. Bestimme für jeden Zustand:

$$v_j^+(i) = \min_{u \in U(i)} \left(E(g(i,u) + \alpha \cdot \sum_{k=1}^{m} p(i,u,k) v_{j-1}^+(k) \right)$$

   und speichere gewählte Politik in $\boldsymbol{\sigma}^+$.
3. Falls für vorgegebenes ε>0:
   $| (v^+_j(i) - v^+_{j-1}(i))/ v^+_j(i)| < \varepsilon$ für alle i = 1,...,m,
   setze $\mathbf{v}^+ = \mathbf{v}^+_j$ und $\boldsymbol{\sigma}^+ = \boldsymbol{\sigma}^+_j$,
   ansonsten setze j=j+1 und fahre bei 1. fort

Einige Bemerkungen zum Verhalten des Algorithmus:

- Die optimale Politik muss nicht erreicht werden
- Auch wenn das Abbruchkriterium erfüllt ist, kann die gefundene Lösung noch weit vom Optimum entfernt sein.
- Die Iterationen sind sehr effizient realisierbar, da keine Gleichungssysteme zu lösen sind.

Politikiteration:

Idee (auf Basis des Wissens, dass eine optimale Politik existiert):

1. starte mit einer Politik $\boldsymbol{\sigma} = (\sigma(1),...,\sigma(m))$
   wähle eine gute Näherungspolitik oder
   wähle $\sigma(i)$ so dass $E(g(i,u_{i\sigma(i)}))$ minimal
2. berechne die zugehörigen Werte $\mathbf{v} = (v(1),...,v(m))$
   $\mathbf{v}$ ist die Lösung des linearen Gleichungssystems

$$v(i) = E(g(i,\sigma(i))) + \alpha \cdot \sum_{k=1}^{m} p(i,\sigma(i),k)v(k) \quad (i = 1,\ldots,m)$$

3. bestimme verbesserte Politik $\boldsymbol{\sigma}$‘ mit zugehörigen Werten $\mathbf{v}$‘ ($\leq\mathbf{v}$)
   neue Politik $\boldsymbol{\sigma}$‘ wird aus der folgenden Minimierung bestimmt

$$\sigma'(i) = \underset{u\in U(i)}{\operatorname{argmin}} \left(E(g(i,u)) + \alpha \cdot \sum_{k=1}^{m} p(i,u,k)v(k) \right)$$

4. falls $\boldsymbol{\sigma}$‘=$\boldsymbol{\sigma}$ wurde die optimale Politik $\boldsymbol{\sigma}^{+}$ erreicht, ansonsten
   setze $\boldsymbol{\sigma}$=$\boldsymbol{\sigma}$‘ und fahre bei 2. fort

Einige Bemerkungen zum Abschluss

- Politikiteration sollte verwendet werden, wenn
  - eine gute initiale Politik bekannt ist
  - der Zustandsraum relativ klein ist
- Werteiteration sollte verwendet werden, wenn
  - der Zustandsraum sehr groß ist
  - der resultierende Markov-Prozess nicht (immer) irreduzibel ist
- Weiter Möglichkeiten
  - Lineare Programmierung
    (als Erweiterung der Vorgehensweise bei der Berechnung erwarteter Kosten)
  - Hybride Ansätze als Mischung von Wert- und Politikiteration

12.7 Optimales Stoppen von Markov Ketten

Typische Entscheidungssituation:

Kaufen/Verkaufen einer Ware oder

Warten auf zukünftige Angebote

oft ist das Warten mit Kosten verbunden

Typische Anwendungsgebiete:

- Finanzmathematik
- Kaufentscheidungen
- Sequentielle Entscheidungen

Formale Darstellung:

Markovscher Entscheidungsprozess mit Zustandsraum $X=\{0,\ldots,m\}$

Zustand 0 Stoppzustand, $X' = X \setminus \{0\}$

$U(i) = \{S,W\}$ für $i \in X'$ mit S stoppen, W weitermachen

$U(0) = \{S\}$

Übergangswahrscheinlichkeiten $p(i,u,k)$ mit

- $p(i,S,0) = 1$ für alle $i \in X$
- $p(i,W,0) = 0$ für alle $i \in X'$

Kosten

- $c_j(i)$ bei Wahl von W
- $s_j(i)$ bei Wahl von S in $i \in X'$ und $s_j(0) = 0$
- Bei endlichen Horizonten finale Kosten $g_n(i)$ für alle $i \in X'$

Bellmansche Funktionsgleichungen

$$E(V_j^*(i)) = \begin{cases} \min\left(c_j(i) + \sum\limits_{k \in X'} p(i,W,k) E(V_j^*(k)), s_j(i) + E(V_{j+1}^*(0)\right) & \text{falls } j < n \wedge i \neq 0 \\ E(V_{j+1}^*(0)) & \text{falls } j < n \wedge i = 0 \\ g_n(i) & \text{falls } j = n \end{cases}$$

Der Markov Kette wird gestoppt, wenn im ersten Punkt, das Minimum durch den zweiten Term erreicht wird

Rückwärtsberechnung bis alle Stoppzeiten bestimmt oder Periode 1 erreicht

Beispiel:

Es soll eine Ware innerhalb von n Perioden verkauft werden

Es werden in jeder Periode Gebote aus der Menge $\{w_1,\ldots,w_m\}$ abgegeben

Falls die Ware nicht verkauft wird, so müssen Fixkosten K pro Periode aufgebracht werden

Wenn in Periode j Gebot w_i abgegeben wurde, so wird mit Wahrscheinlichkeit p(i,k) in Periode j+1 Gebot w_k abgegeben

Formalisierung:

- $P(i,W,k) = p(i,k)$
- $c_j(i) = K$, $s_j(i) = -w_i$, $g_n(i) = -w_i$

Falls keine Zeitschranke für das Stoppen vorgegeben wird, so nehmen wir an, dass

- c(i) und s(i) unabhängig von der Periode sind
- $s(i) < \infty$ und $0 \leq c(i) < \infty$ und von jedem Zustand $i \in X'$ ein Zustand $k \in X'$ mit $c(i) > 0$ erreichbar ist, wenn Entscheidung W getroffen wird

Für die Kosten $v^*(i)$ dann gilt dann

$$v^*(i) = \min \left(s(i), c(i) + \sum_{k \in X'} p(i, W, k) \cdot v^*(k) \right)$$

Berechnung per Werte- oder Politikiteration