8 Dynamische Optimierung

Motivation/Einführung: Dynamische Optimierung
Sequentielle Optimierung
Stufenoptimierung

zwei mögliche Pfade

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

- sich „im Lauf der Zeit“ offenbaren ⇒ „dynamisch“ zu berücksichtigen sind, z.B. Lagerhaltung über mehrere Perioden
  ⇒ stochastische Techniken fließen auf natürliche Weise ein
  (tatsächliche Unsicherheiten)

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

- „im Laufe des Verfahrens“ (Divide & Impera) vervollständigen ⇒ sequentiell arbeiten,
  z.B. Knapsack Problem mit Entscheidungsbaum ⇒
  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

Gliederung

8.1 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
- Es resultiert Lagerbestand x_j zu Ende Periode j, unmittelbar vor Periode j+1, als Bestand zu Periode j (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 $g(x_j,u_j)$
- 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:

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

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

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

$x_j \geq 0$ und $u_j \geq 0 \quad$ 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: $h_E(x_{j-1},u_j)$ Beschaffung - Restwert
  - Wartung: $h_W(x_{j-1},u_j)$
  - Gesamt: $g(x_j,u_j)$
- Optimierungsaufgabe:
  - bestimme Ersatzzeitpunkte j (Entscheidungsvariable u_j)
  - unter Einhaltung der Nebenbedingungen / Annahmen

  derart, dass Gesamtkosten minimal werden.

Formal:

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

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

$$x_1 = 1$$

$$x_j \in \{1,2,...,j\} \quad \text{für } j=2,...,n$$

$$u_j \geq 0 \quad \text{für } j=1,2,...,n$$

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 $a_{max} > 0$

gepackte Werte-Summe zu maximieren

Entscheidungsvariablen (hier)

$u_j=1$ Gegenstand j wird eingepackt

$u_j=0$ Gegenstand j wird nicht eingepackt

Formal:

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

„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: $x_1=a_{max}$

Formal:

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

udN

$$x_{j+1} = x_j - a_j u_j \text{ für } j = 1,2,...,n \quad \text{und} \quad x_1 = a_{max}$$

$$0 \le x_{j+1} \le a_{max}$$

$$u_j \in \{0,1\} \text{ für } x_j \ge a_j \quad \text{und} \quad u_j = 0 \quad \text{für } x_j < a_j$$

8.2 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 Planungszeitraum 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 $U_j(x_j)$, $u_j \in U_j(x_j) \neq \emptyset$ (abhängig von Vorzustand)
- in Form erlaubter (nichtleerer) Zustandsbereiche X_{j+1}, $x_{j+1} \in X_{j+1} \neq \emptyset$ 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 (dementsprechend) erklärt auf $D_j := \{(x,u) | \, x \in X_j, \, u \in U_j(x)\}$
- Optimierungsaufgabe besteht aus Minimierung Kosten (Maximierung Erlösen) über gesamten Planungszeitraum
  - hier vorrangig Kostenminimierung betrachtet

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

kann sehr komplex sein! $\quad x_{j+1} \in X_{j+1}$ und $u_j \in U_j(x_j)$ für $j = 1,2,...,n$

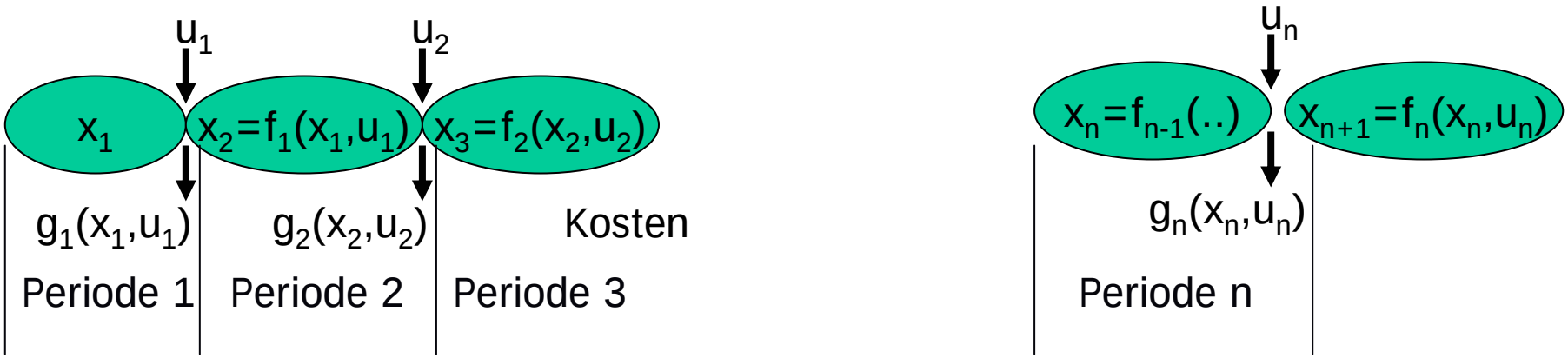
Struktur der vorgestellten Beispiele

	Beispiel 1 Lagerhaltung	Beispiel 2 Erneuerung	Beispiel 3 Rucksack
x_j	reellwertig Bestand	ganzzahlig Alter	reellwertig Kap.
X_j	$\mathbb{R}_+$ für j=2,..,n $X_0=X_{n+1}=\{0\}$	{1,...,j-1} für j=1,..,n+1, $X_1=\{0\}$	$[0,a_{max}]$ für j=2,..,n+1, $X_1=\{a_{max}\}$
u_j	reellwertig	binär	binär
$U_j(x_j)$	$\mathbb{R}_+$	{0,1}	{0,1}
$f_j(x_j,u_j)$	$x_j + u_j - r_j$	$x_j(1 – u_j) + 1$	$x_j - a_ju_j$
$g_j(x_j,u_j)$	$h_B(u_j) + h_Lx_j$	$h_E(u_j,x_j) + h_W(u_j,x_j)$	$-c_ju_j$
	Kosten	Kosten	Nutzen (zu maximieren!)

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 „Markov´sch“ 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!

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 8.1 (Bellman´sches 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 **Bellman'sche Funktionsgleichung**:

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

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)

8.3 Bellman‘sche Funktionsgleichungsmethode

In direkter Anwendung der B´schen Funktionalgleichung, naheliegende Vorgehensweise (Alternativen existieren)

- 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 de 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 Rückwartsphase 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

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

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 $X_5=\{0\}$

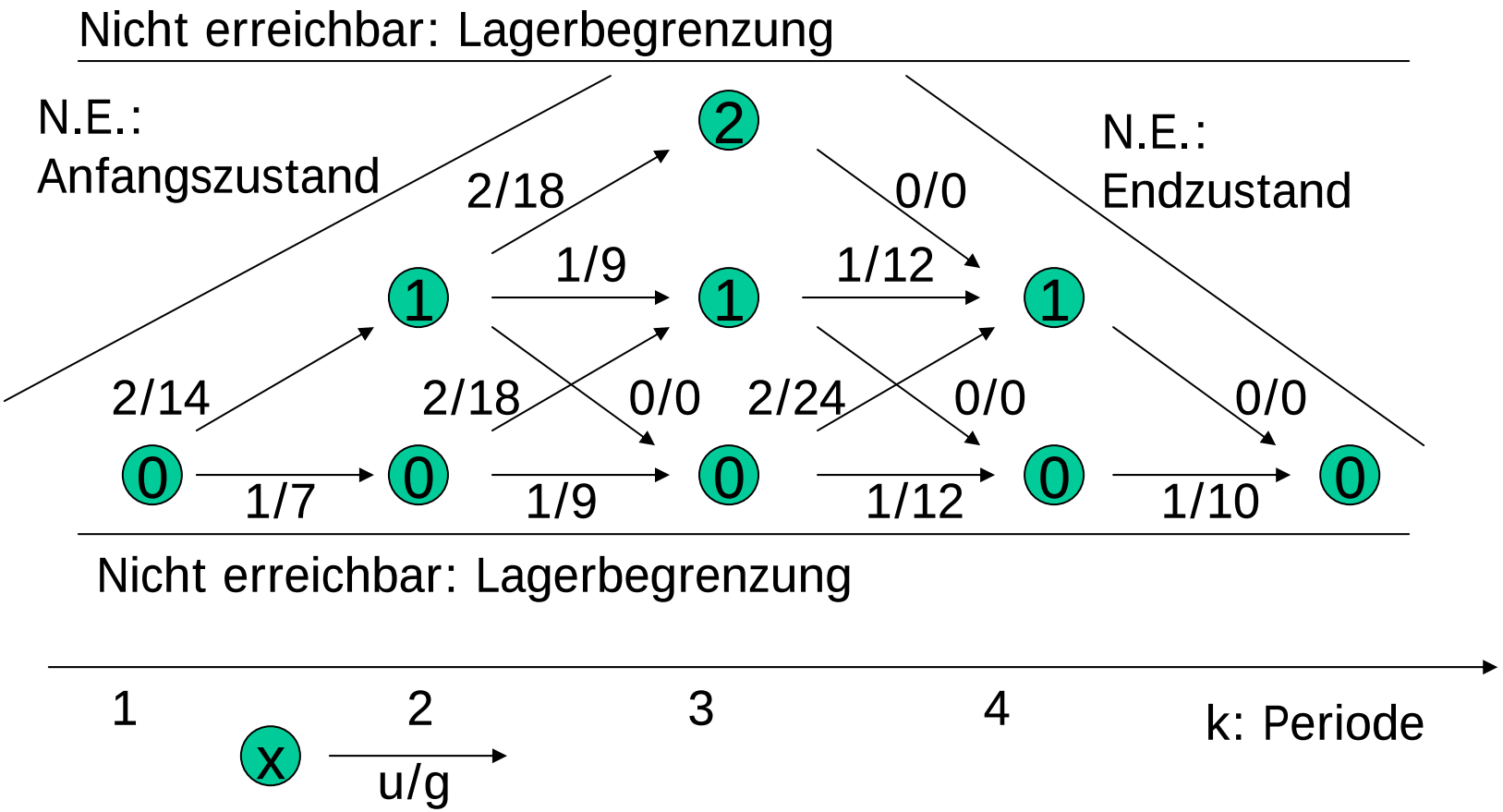
I.a. tritt ganz X als Menge möglicher Endzustände auf!
Randbed: $v^*_{n+1}(x_{n+1})=0$

- 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 attributierten Digraphen möglich
- Darstellung der (diversen) Funktionen im Verfahrensverlauf mittels Tabellen sinnvoll

Veranschaulichung des Problemraums



Durchführung: Bellman'sche 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		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)

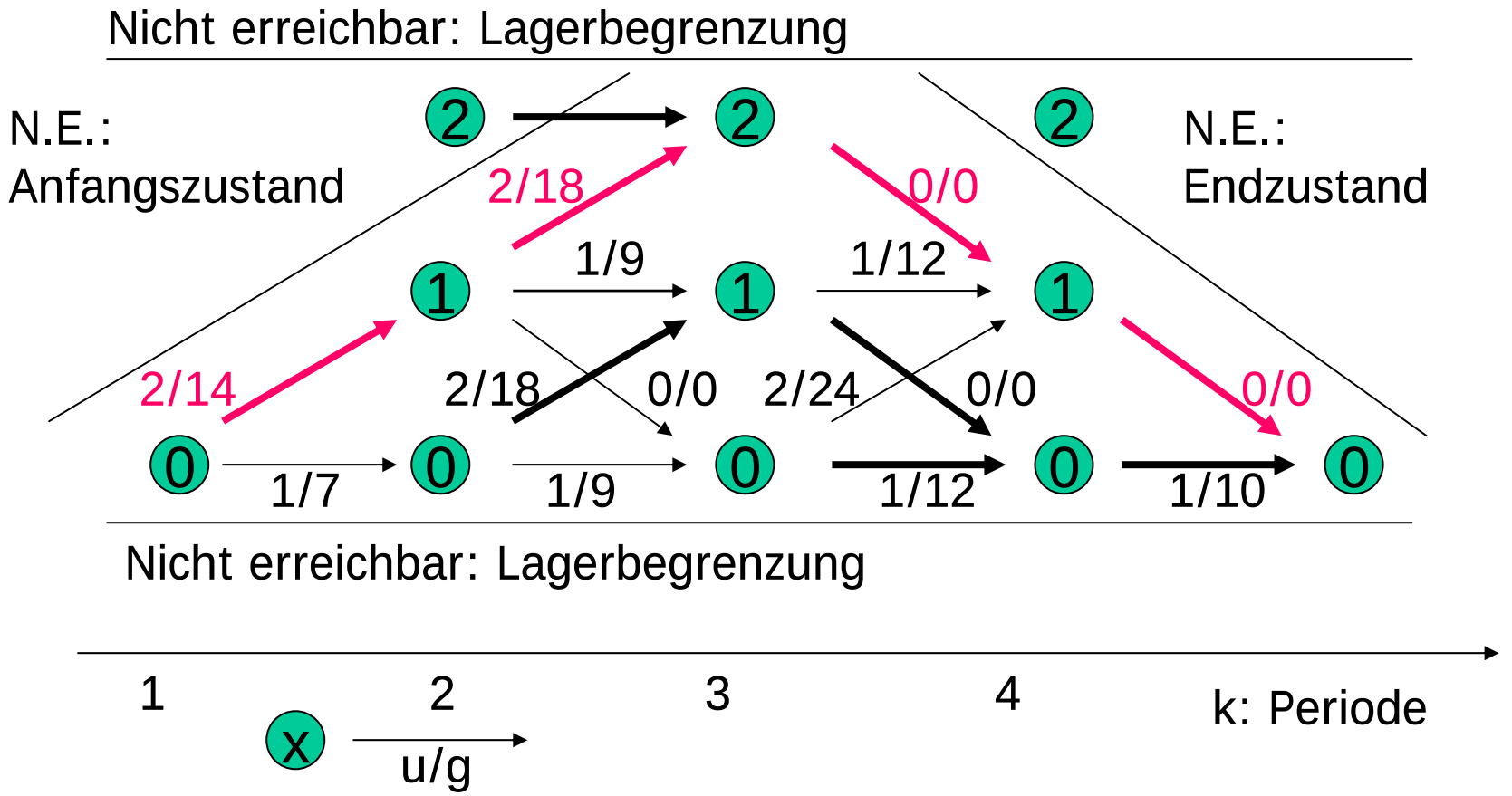
$\Rightarrow$ optimales Ergebnis $v^*_1(0) = 32$

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 $0 \leq u_j \leq x_j$
- 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 $g_j(x_j,u_j) = \sqrt{u_j}$ für 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 $g_j(x_j,u_j) = \sqrt{u_j}$ für j=1,2,3

Im Gegensatz zu den bisherigen Beispielen kontinuierlicher Zustandsraum und Entscheidungsraum!

Durchführung Bellman'sche Funktionalgleichungsmethode angepasst an „Maximierung" und Problem

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

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

Einige Vorüberlegungen:

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

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

w(u) strikt konkav über [0,x]

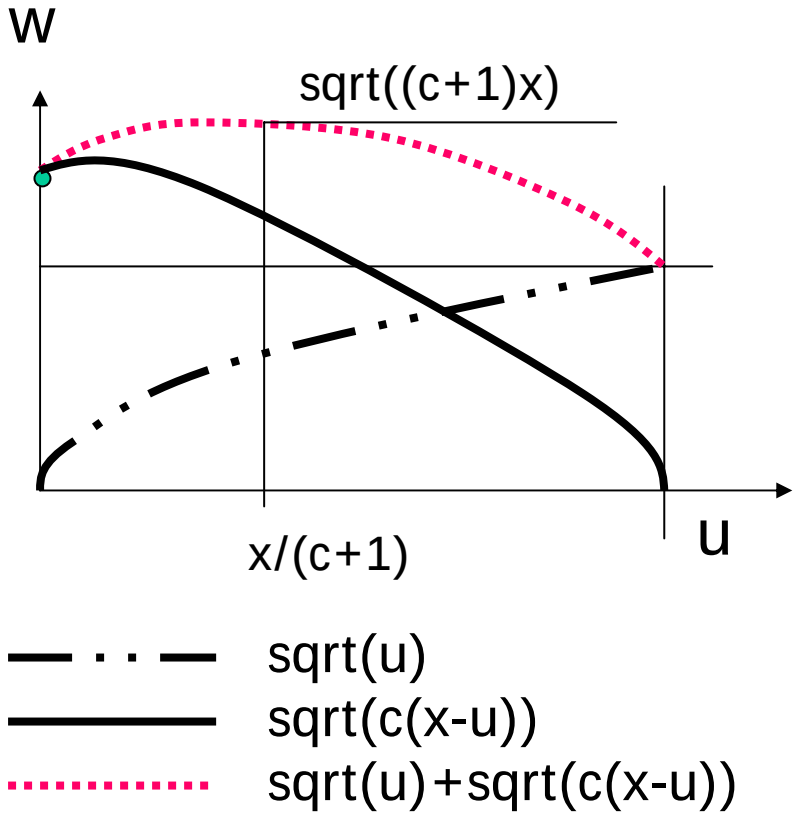
=> Extremstelle in [0,x] ist Maximum

Bestimmung der Extremstelle durch Nullsetzen der ersten Ableitung:

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

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

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



Rückwärtsentwicklung:

Periode 3:

- vereinbarungsgemäß ist $v^*_4(x_4)=0$
- und daher $v^*_3(x_3)=\max_{0\le u_3\le x_3}\{\sqrt{u_3}+0\}$
- $v^*_3(x_3)=\sqrt{x_3}$ $z^*_3(x_3)=x_3$
  (zu Ende alles verkaufen)

Periode 2:

- $v^*_2(x_2)=\max_{0\le u_2\le x_2}\{\sqrt{u_2}+\sqrt{[2(x_2-u_2)]}\}$
- Maximalstelle ?
  aus Vorüberlegungen, für c=2,
  $v^*_2(x_2)=\sqrt{3}x_2$ $z^*_2(x_2)=x_2/3$

Periode 1:

- $v^*_1(x_1)=\max_{0\le u_1\le x_1}\{\sqrt{u_1}+\sqrt{[6(x_1-u_1)]}\}$
- Maximalstelle ?
  aus Vorüberlegungen, für c= 6
  $v^*_1(x_1)=\sqrt{7}x_1$ $z^*_1(x_1)=x_1/7$

Vorwärtsentwicklung:

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

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

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

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

$\Rightarrow$ 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 pseudopolynomial: 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)

8.4 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 Z_j mit $x_j \in Z_j \neq \emptyset$ (Benennungsänderung $X \rightarrow Z$) mit bekanntem Anfangszustand $x_1=x_a$ und $Z_1=\{x_1\}$
- 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
(im Rahmen der Modellbeschreibung)

- nicht zu eindeutigem Folgezustand x_{j+1}
- sondern zu einem der erlaubten Folgezustände $x_{j+1} \in Z_{j+1}$

Motivierende Betrachtungen

In Bsp 1 könnten

- Auslieferungsmengen statt konstant 1 als schwankend angenommen sein, wobei die Ursache des Schwankens
  - entweder tatsächlich unbekannt (Unsicherheit)
  - oder nicht detailliert beschrieben (Aufwandsreduktion)
- Belieferungskosten statt je Periode j zu q_j €/Stück fest als schwankend angenommen sein, wobei die Ursache des Schwankens
  - entweder tatsächlich unbekannt (Unsicherheit)
  - oder nicht detailliert beschrieben (Aufwandsreduktion)

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}, deren Verteilung die „Regelmäßigkeiten des Schwankens“/Häufigkeiten/Wahrscheinlichkeiten der realisierten Folgezustände $x_{j+1} \in Z_{j+1}$ beschreibt
  - wo bei endlichem / abzählbarem Zustandsbereich Z_{j+1} Verteilung charakterisiert (z.B.) durch Menge bedingter Wahrscheinlichkeiten
  - wo bei kontinuierlichem Zustandsbereich Z_{j+1} Verteilung charakterisierbar (z.B.) durch (bedingte) Dichtefunktion
- werden resultierende Kosten/Erlöse erfasst durch erweiterte Funktion $g_j(x_j,u_j,x_{j+1})$, d.h. zusätzlich von realisiertem Folgezustand abhängig)
  - 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})

Bellman‘sche Funktionalgleichung

Unter diesen Annahmen (Ziel: Bellman‘sche 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 $g_j(x_j,u_j,x_{j+1})+E[V^*_j(x_{j+1})]$
  (Folgezustände treten gemäß Verteilung auf)
- ergibt sich die Bellman‘sche Funktionalgleichung analog zur Ableitung der deterministischen Form bei diskreter Zustandsbereich Z_{j+1}

$$E[V^*_j(x_j)] = \min_{u_j \in U_j(x_j)} \sum_{x \in Z_{j+1}} \left\{ \left(g_j(x_j,u_j,x) + E[V^*_{j+1}(x)] \right) PX_{j+1}(x \mid x_j,u_j) \right\}$$

Spezialfall der stochastisch dynamischen Entscheidungsprozesse:

Markov‘sche Entscheidungsprozesse mit endlichem Zustandsraum und Steuerbereich

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

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

Sei x_j=i Zustand zu Beginn von Periode j

- System geht zu Beginn von Periode j+1 mit Wahrscheinlichkeit $p_{ik}(u_{i\sigma})$ in Zustand x_{j+1}=k bei Entscheidung $u_{i\sigma}$ über
- es gilt für die Übergangswahrscheinlichkeiten $p_{jk}(u_{i\sigma})$:
  $\sum_{k=1}^{m} p_{ik}(u_{i\sigma}) = 1.0$ für alle $i \in X$ und $u_{i\sigma} \in U(i)$
- Übergangswahrscheinlichkeiten hängen nur vom Zustand und der gefällten Entscheidung ab
  (homogener Markov‘scher Entscheidungsprozess)
- Erwartete Kosten der Entscheidung $u_{i\sigma}$ in Zustand i:

$$E(g(i,u_{i\sigma})) = \sum_{k=1}^{m} p_{ik}(u_{i\sigma}) g(i,u_{i\sigma},k)$$

$g(i,u_{i\sigma},k)$ Kosten, die anfallen, wenn in Zustand i Entscheidung $u_{i\sigma}$ zu einem Zustandsübergang nach k führt

Belman‘sche Funktionsgleichung:

$$v^*_j(i) = \min_{\sigma=1,\ldots,s_i} \left\{ E(g(i,u_{i\sigma})) + \sum_{k=1}^{m} p_{ik}(u_{i\sigma}) v^*_{j+1}(k) \right\}$$

Sei $\sigma^*_j(i)$ ein Entscheidung σ, für den das Minimum $v^*_j(i)$ angenommen wird

(also eine optimale Entscheidung im Zustand i für Periode j)

Bestimmung von $\sigma^*_j(i)$ und $v^*_j(i)$:

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

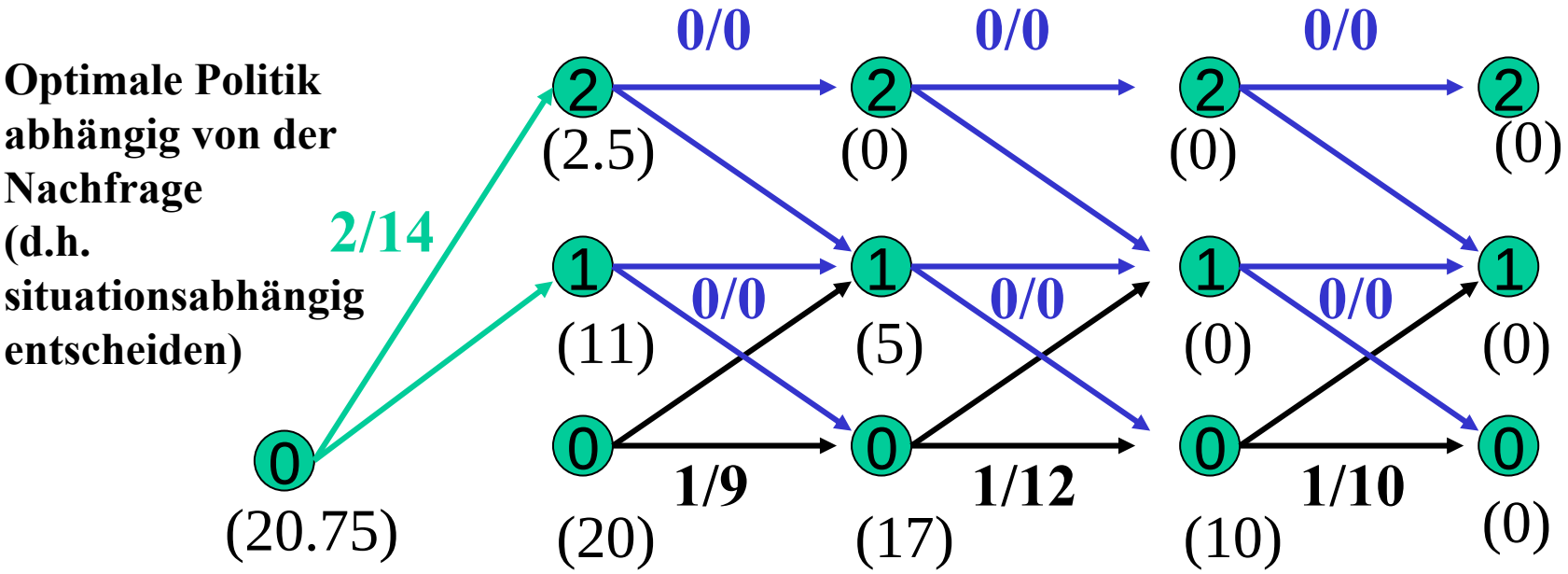
Vorgehen im Prinzip analog zum deterministischen Fall

Erweiterung des Beispiels von Folie 21-23

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



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

Oft sollen Gewinne/Kosten über einen unendlichen Planungshorizont untersucht werden, dann ist 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 $\alpha < 1$ geschmälert d.h. Gewinn x in k Perioden ist jetzt nur $\alpha^k \cdot x$ wert

Analysemethoden:

- Lineare Programmierung
- Politikiteration

Bestimmung des Erwartungswerts pro Schritt:

- $v^*_j(i)$ sind die minimalen Kosten im Zustand i nach j Schritten
- $c_j(i) = v^*_j(i) / j$

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 $E(C^+)$ sind die minimalen erwarteten Kosten des Markov'schen Entscheidungsproblems mit unendlichem Planungshorizont sind
- eine stationäre optimale Politik $\sigma^+ = (\sigma^+(1),...,\sigma^+(m))$ 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 σ^+ erfüllen die Gleichung $E(C) = \sum_{k=1}^{m} \pi(k) E(g(i, u_{i\sigma^+(i)}))$

mit $\pi(i) = \sum_{k=1}^{m} \pi(k) p_{ki}(u_{k\sigma^+(k)})$ und $\sum_{k=1}^{m} \pi(k) = 1$ stationäre Lösung

Annahme stationäre Lösung existiert ($\Rightarrow$ irreduzibler Markov-Prozess)

Darstellung als lineares Programm:

Sei $y_{i\sigma}$ ($i \in \{1,...,m\}$, $\sigma \in \{1,...,s_i\}$) die Wahrscheinlichkeit im Zustand i zu sein und Entscheidung σ zu treffen

Es gilt (lineare Nebenbedingungen):

$$\sum_{i=1}^{m} \pi(i) = \sum_{i=1}^{m} \sum_{\sigma=1}^{s_i} y_{i\sigma} = 1.0 \text{ und}$$

$$\sum_{\sigma=1}^{s_i} y_{i\sigma} = \sum_{i=1}^{m} \sum_{\sigma=1}^{s_i} y_{i\sigma} p_{ij}(\sigma)$$

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

Zielfunktion: $\min \sum_{i=1}^{m} \sum_{\sigma=1}^{s_i} y_{i\sigma} E(g(i, u_{i\sigma}))$

Mittels Simplex-Algorithmus lösbar

Variablenzahl:

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

Aus dem bisher gesagten folgt, dass für die optimale Lösung

$y_{i\sigma} = 1$ für genau ein σ und für alle anderen $\rho \in \{1,...,s_i\} \setminus \{\sigma\}$ $y_{i\rho}=0$

Lösung mittels Politikiteration:

Für eine Politik $\boldsymbol{\sigma} = (\sigma(1),..,\sigma(m))$ gilt

$$G(\boldsymbol{\sigma}) + v(i) = E(g(i, u_{i\sigma(i)})) + \sum_{k=1}^{m} p_{ik}(u_{i\sigma(i)})v(k) \quad (i = 1,...,m)$$

wobei $G(\boldsymbol{\sigma}) = \sum_{k=1}^{m} E(g(k, u_{k\sigma(k)}))\pi(k)$ und $\pi(k) = \sum_{i=1}^{m} p_{ik}(u_{i\sigma(i)})\pi(i)$

<u>Erläuterung des Zusammenhangs für eine feste Politik $\boldsymbol{\sigma}$</u>

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

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):

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

„Ausprobieren“ aller möglichen Politiken führt zu einem Algorithmus mit Aufwand Πs_i (inakzetabel)
Besser schrittweise Verbesserung der Politik

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

$$G(\boldsymbol{\sigma}) = E(g(i, u_{i\sigma(i)})) + \sum_{k=1}^{m} p_{ik}(u_{i\sigma(i)})v(k) - v(i) \quad (i = 1,..., m)$$

   wobei $v(m) = 0$ gesetzt wird (sonst keine eindeutige Lsg.)
3. bestimme verbesserte Politik $\boldsymbol{\sigma}$‘ aus der folgenden Minimierung

$$\min_{\sigma'(i)=1,...,s_i} \left(E(g(i, u_{i\sigma'(i)})) + \sum_{k=1}^{m} p_{ik}(u_{i\sigma'(i)})v(k) \right) \quad \text{für } i = 1,...,m$$

4. Falls $\boldsymbol{\sigma}$‘=$\boldsymbol{\sigma}$ optimale Politik erreicht, sonst $\boldsymbol{\sigma}$=$\boldsymbol{\sigma}$‘ und fahre bei 1. fort (es gilt dann $G(\boldsymbol{\sigma}') < G(\boldsymbol{\sigma})$)

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

Bellman‘sche Funktionsgleichungen:

$$v^{+}{}_{j}(i) = \min_{\sigma=1,\ldots,s_i} \left\{ E(g(i,u_{i\sigma})) + \alpha \cdot \sum_{k=1}^{m} p_{ik}(u_{i\sigma}) v^{+}{}_{j-1}(k) \right\}$$

mit $v^{+}{}_{0}(i) = 0$ für $i = 1,\ldots, m$

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

Wertiteration:

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 Markov'schen 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 $\mathbf{v}^+ = (v^+_1,...,v^+_m)$ auftreten
- $\mathbf{v}^+$ und $\boldsymbol{\sigma}^+$ erfüllen die Gleichungen

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

Werteiteration (schrittweise Approximation der optimalen Kosten):

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

$$v_n^+(i) = \min_{\sigma_n^+(i)\in\{1,\ldots,s_i\}} \left(E(g(i, u_{i\sigma_n^+(i)})) + \alpha \cdot \sum_{k=1}^{m} p_{ik}(u_{i\sigma_n^+(i)}) v_{n-1}^+(k) \right)$$

   und speichere gewählte Politik σ^+.

3. Falls für vorgegebenes $\varepsilon>0$:
   $|(v^+_n(i) - v^+_{n-1}(i))/ v^+_n(i)|<\varepsilon$ für alle i = 1,...,m,
   setze $\mathbf{v}^+ = \mathbf{v}^+_n$ und $\sigma^+ = \sigma^+_n$

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

$$v(i) = E(g(i,u_{i\sigma(i)})) + \alpha \cdot \sum_{k=1}^{m} p_{ik}(u_{i\sigma(i)})v(k) \quad (i = 1,...,m)$$

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

$$E(g(i,u_{i\sigma'(i)})) + \alpha \cdot \sum_{k=1}^{m} p_{ik}(u_{i\sigma'(i)})v(k) =$$
$$\min_{\sigma=1,...,s_i} \left(E(g(i,u_{i\sigma})) + \alpha \cdot \sum_{k=1}^{m} p_{ik}(u_{i\sigma})v(k) \right)$$

Falls $\boldsymbol{\sigma}$‘=$\boldsymbol{\sigma}$ wurde die optimale Politik $\boldsymbol{\sigma}^+$ erreicht $\Rightarrow$
$\boldsymbol{\sigma}^+$ wird immer nach endlich vielen Iterationen erreicht

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