



Universiteit
Leiden
The Netherlands

Casinomodellen

Fung, M.

Citation

Fung, M. (2007). *Casinomodellen*.

Version: Not Applicable (or Unknown)

License: [License to inclusion and publication of a Bachelor or Master thesis in the Leiden University Student Repository](#)

Downloaded from: <https://hdl.handle.net/1887/3596872>

Note: To cite this publication please use the final published version (if applicable).

Casinomodellen

Michael Fung

Bachelorscriptie - Voorjaar 2007

Inleiding

Deze bachelorscriptie is geschreven in het kader van het Bachelor Seminarium dat aan de Universiteit van Leiden werd gegeven tijdens het tweede semester van het studiejaar 2006/2007. Voor deze scriptie heb ik onder begeleiding van Prof. L.C.M Kallenberg een besliskundig probleem moeten onderzoeken.

Het probleem luidt grofweg als volgt: een speler gaat met een bepaald startbedrag naar het casino en wil na afloop met een vooraf met zichzelf afgesproken bedrag weer naar huis gaan. De vraag is dan welke strategie de speler het beste kan hanteren. In hoofdstuk 1 zal het probleem in details geformuleerd worden.

Om het casinoprobleem aan te pakken is het handig dat het wordt omgezet in een zogenaamde Markov beslissingsmodel. In hoofdstuk 2 zal worden uitgelegd wat dit precies is en vervolgens wordt in hoofdstuk 3 het probleem omgezet in dit model.

De beste strategie voor de speler blijkt af te hangen van een zogenaamde winkans p . Er zal een onderscheid moeten worden gemaakt tussen drie situaties: $0 < p < \frac{1}{2}$, $p = \frac{1}{2}$ en $\frac{1}{2} < p < 1$. De situaties $0 < p < \frac{1}{2}$ en $\frac{1}{2} < p \leq 1$ worden in de hoofdstukken 4 en 5 behandeld.

Als $0 < p < \frac{1}{2}$ dan is het probleem wat lastiger op te lossen. Vandaar dat dit deelprobleem onderverdeeld is in vier hoofdstukken, namelijk de hoofdstukken 6, 7, 8 en 9.

Vervolgens zijn een aantal simulaties uitgevoerd in hoofdstuk 10. En tenslotte wordt deze scriptie afgesloten met een conclusie.

In deze scriptie zal het symbool \mathbb{N} voor de natuurlijke getallen een aantal keer worden gebruikt. In dit werk heeft dit symbool de volgende betekenis: $\mathbb{N} = \{1, 2, 3, \dots\}$. De natuurlijke getallen met het element 0 eraan toegevoegd zal dan worden genoteerd met \mathbb{N}_0 , dus $\mathbb{N}_0 = \{0, 1, 2, \dots\}$.

Inhoudsopgave

1	Het probleem	3
2	Markov beslissingsketen	4
3	Casinomodel	5
4	Transiënt	7
5	Verdisconteerd model	10
6	Monotone contractie afbeeldingen	15
7	Optimaliteitsvergelijking	20
8	Successieve approximatie	24
9	De winkans is kleiner dan een half	26
10	Simulatie	33
11	Conclusie	38

1 Het probleem

Een speler gaat het casino in en start met $B \in \mathbb{N}$ euro's. Hij speelt daar herhaald hetzelfde spel. Bij dit spel kunnen alleen hele euro's worden ingezet en is er een winkans $p \in (0, 1)$ en een verlieskans $1 - p$. Bij winst krijgt hij zijn inzet verdubbeld terug en bij verlies is hij zijn inzet kwijt. Het doel van de speler is om een van te voren bepaald bedrag van N euro's, met $N > B$, te bereiken. De speler stopt als hij dit streefbedrag N heeft bereikt of als hij blut is.

De vraagstelling luidt: welke strategie kiest de speler, d.w.z. welke inzet a kiest hij als hij i euro's bezit met $a \in \{1, 2, \dots, i\}$, om de kans om het spel te eindigen met het streefbedrag N te maximaliseren?

Om dit probleem op te lossen moet ik de volgende opdrachten uitvoeren:

1. Modelleer het probleem als een Markov beslissingsketen.
2. Toon aan dat dit model een zogenaamd transiënt model is en stel de optimaliteitsvergelijking op.
3. Toon aan dat de optimaliteitsvergelijking een unieke oplossing heeft.
4. Bepaal de optimale strategie voor:
 - (a) $0 < p < \frac{1}{2}$;
 - (b) $p = \frac{1}{2}$;
 - (c) $\frac{1}{2} < p < 1$.
5. Voer een aantal simulaties uit.

2 Markov beslissingsketen

In dit hoofdstuk zullen we definiëren wat een Markov beslissingsketen is. Zij S een niet-lege eindige toestandsruimte, zeg $S = \{1, 2, \dots, N\}$. Bij iedere toestand $i \in S$ hoort een niet-lege eindige actieverzameling $A(i)$. Op ieder beslissingstijdstip t , met $t = \{0, 1, 2, \dots\}$, bevindt het systeem zich in één van de toestanden. Als het systeem zich op een zeker tijdstip in toestand i bevindt, wordt een actie $a \in A(i)$ gekozen. Als in toestand i de actie a wordt gekozen, dan is er een directe opbrengst $r_i(a) \in R$. Verder is het systeem op het volgende beslissingstijdstip in toestand j met overgangskans $p_{ij}(a)$ als i de huidige toestand is en actie a wordt gekozen. Dit model kan dan worden weergegeven met een aantal Markov ketens. Het aantal Markov ketens dat je nodig hebt is gelijk aan $\prod_{k=1}^N \#A(k)$, want in iedere toestand moet een actie worden gekozen. Dus het aantal Markov ketens is gelijk aan het product van het aantal elementen dat de afzonderlijke $A(i)$'s hebben.

Zij b^t de beslisregel op tijdstip t . Een beslisregel b^t mag afhangen van de toestanden op de tijdstippen $0, 1, 2, \dots, t$ en van de acties op de tijdstippen $0, 1, \dots, t-1$. Dus b^t hangt in het algemeen af van alle informatie die het systeem tot tijdstip t heeft verkregen. Laat

$$S \times A = \{(i, a) | i \in S, a \in A(i)\},$$

dus $S \times A$ is de verzameling van alle paren met als objecten een toestand en een element uit de actieverzameling die bij die toestand hoort. En laat H_t de verzameling zijn van de mogelijke histories van het systeem, dus

$$H_t = \{i_0, a_0, i_1, a_1, \dots, i_{t-1}, a_{t-1}, i_t | (i_k, a_k) \in S \times A, 1 \leq k \leq t-1; i_t \in S\}.$$

De formele definitie van beslisregel b_t luidt dan: een beslisregel b_t op tijdstip t geeft de kans, als functie van H_t naar $A(i_t)$, om een bepaalde actie te kiezen. Dus er moet gelden:

$$b_{h_t a_t}^t \geq 0 \text{ voor alle } a_t \in A(i_t) \text{ en } \sum_{a_t} b_{h_t a_t}^t = 1 \text{ voor alle } h_t \in H_t.$$

Een strategie R is dan een rij beslisregels, dus $R = (b^0, b^1, b^2, \dots)$. Een strategie R wordt geheugenloos genoemd als alle beslisregels b^t , met $t \in \mathbb{N}_0$, niet afhangen van het verleden. Dus b^t mag alleen afhangen van de toestand i_t op tijdstip t . In dit geval wordt de notatie dan ook $b_{i_t a_t}^t$ i.p.v. $b_{h_t a_t}^t$. Een beslisregel wordt deterministisch genoemd als er geldt: $b_{i_t a_t}^t \in \{0, 1\}$. Dit houdt dus in dat de beslisregel alleen afhankelijk is van de actie in de toestand op tijdstip t die met kans 1 wordt gekozen. In dit geval wordt de beslisregel genoteerd als een functie $f_t : S \rightarrow A$. Een strategie die alleen maar deterministische beslisregels bevat wordt dan ook een deterministische strategie genoemd. Een geheugenloze deterministische strategie wordt genoteerd als $R = (f_0, f_1, f_2, \dots)$. Verder wordt een (geheugenloze deterministische) strategie stationair genoemd als alle beslisregels identiek zijn (dus $f_0 = f_1 = f_2 = \dots$) en noteren we de strategie met f^∞ .

3 Casinomodel

We zullen in dit hoofdstuk zien hoe het casinoprobleem omgezet kan worden in een Markov beslissingsketen. Laat B het bedrag in euro's zijn waarmee de speler begint, N het streefbedrag, en p de winstkans van een spel (dus de verlieskans is $1 - p$). Als toestandsruimte S nemen we het aantal euro's dat de speler in bezit heeft, dus er geldt

$$S = \{0, 1, \dots, N\}.$$

En als beslissingstijdstippen nemen we $t = \{0, 1, 2, \dots\}$, corresponderend met het aantal spellen dat geweest is.

Als de speler op een gegeven moment blut is, zal hij natuurlijk niets meer inzetten, hij heeft immers niets meer om in te zetten. Hij zal ook niets meer inzetten als hij N euro in bezit heeft, want dit is het bedrag waarvoor de speler gekomen was. Maar als de speler in bezit is van i euro's, met $0 < i < N$, kan hij maximaal i euro inzetten. Verder zal hij (in het geval $i \geq \frac{1}{2}N$) natuurlijk ook niet meer dan $N - i$ euro's inzetten, want dit is het bedrag dat de speler tekort komt om zijn doel te bereiken. Dus waarom zou hij onnodige risico's riskeren door meer in te zetten. We zullen ook aannemen dat de speler geen pauze neemt, d.w.z. geen spellen overslaat, en altijd minstens 1 euro inzet als $0 < i < N$. Dus de actieverzameling wordt:

$$A(i) = \begin{cases} \{0\} & \text{als } i = \{0, N\} \\ \{1, 2, \dots, i\} & \text{als } 0 < i < \frac{1}{2}N \\ \{1, 2, \dots, N - i\} & \text{als } \frac{1}{2}N \leq i < N \end{cases}$$

Omdat het enige doel van de speler is om het streefbedrag N te halen, kennen we alleen een directe opbrengst 1 toe in de toestand N en 0 in de andere toestanden en acties. Dus we hebben:

$$r_N(0) = 1 \text{ en } r_i(a) = 0 \text{ voor } a \in A(i), i \neq N.$$

De overgangskansen zijn eenvoudig te bepalen. Als je in de toestand 0 of N bent, stopt het proces. Want in toestand 0 kan de speler niets meer inzetten en in toestand N is de speler tevreden en wil hij niet meer spelen. Dus de overgangskansen in deze gevallen worden:

$$p_{ij}(0) = 0 \text{ voor alle } j \in S \text{ en } i \in \{0, N\}.$$

Voor de overige toestanden i geldt dat als de speler a euro inzet, dan is de kans dat toestand $i + a$ wordt bereikt gelijk aan p en de kans dat toestand $i - a$ wordt bereikt $1 - p$. De overige overgangskansen zijn 0, dus voor $a \in A(i)$, $i \notin \{0, N\}$ geldt:

$$p_{ij}(a) = \begin{cases} p & \text{als } j = i + a \\ 1 - p & \text{als } j = i - a \\ 0 & \text{als } j \neq i + a \vee j \neq i - a \end{cases}$$

Omdat er geldt:

$$\sum_j p_{ij}(a) \leq 1 \text{ voor alle } i \in S \text{ en } a \in A(i),$$

hebben we in dit geval te maken met een substochastisch model. Daarentegen wordt een model stochastisch genoemd, als er een gelijkheid zou gelden:

$$\sum_j p_{ij}(a) = 1 \text{ voor alle } i \in S \text{ en } a \in A(i).$$

4 Transiënt

Zij $p_{ij}^t(R)$ de kans dat het systeem op tijdstip t in toestand j is, gegeven dat het systeem op tijdstip $t = 0$ in toestand i en strategie R wordt gebruikt. De strategie R wordt transiënt genoemd als voor alle $i, j \in S$ geldt:

$$\sum_{t=0}^{\infty} p_{ij}^t(R) < \infty.$$

Een Markov beslissingsketen is transiënt als iedere strategie transiënt is. We definiëren nu een rijtje van vectoren $\{y^t, t = 0, 1, \dots\}$ als volgt:

$$y_i^0 := 1 \text{ voor alle } i \in S$$
$$y_i^t := \max_{a \in A(i)} \sum_j p_{ij}(a) y_j^{t-1} \text{ voor alle } i \in S \text{ en } t \in \mathbb{N}.$$

Stelling 4.1

Iedere strategie is transiënt $\Leftrightarrow \max_i y_i^{\#S} < 1$.

Bewijs

Zie pagina 42 van L.C.M. Kallenberg, 1983. *Linear programming and finite Markovian control problems*. Amsterdam: Mathematisch Centrum.

□

Lemma 4.1

Er geldt $0 \leq y_i^t \leq 1$ voor alle $t \in \mathbb{N}$ en $i \in S$.

Bewijs

Met volledige inductie naar t . Voor $t = 0$ klopt de stelling, want $y_i^0 = 1$ (definitie) voor alle $i \in S$.

Stel de stelling is waar voor alle natuurlijke getallen kleiner dan t , dan geldt $0 \leq y_i^{t-1} \leq 1$ voor alle $i \in S$. Volgens de definitie geldt

$$y_i^t = \max_{a \in A(i)} \sum_j p_{ij}(a) y_j^{t-1}.$$

Het is duidelijk dat $y_i^t \geq 0$, want alle $p_{ij}(a)$'s zijn niet negatief. Verder geldt

$$\sum_j p_{ij}(a) y_j^{t-1} \leq \sum_j p_{ij}(a) \leq 1 \text{ voor alle } a \in A(i).$$

Hieruit volgt:

$$y_i^t = \max_{a \in A(i)} \sum_j p_{ij}(a) y_j^{t-1} \leq 1.$$

□

In het vervolg van dit hoofdstuk gaan we uit van het casinomodel. In dit geval kunnen we de definitie voor $\{y^t, t = 0, 1, \dots\}$ wat vereenvoudigen, omdat voor vele $p_{ij}(a)$'s geldt dat deze gelijk is aan 0. Er geldt nog steeds:

$$y_i^0 := 1 \text{ voor alle } i \in S.$$

Verder geldt voor $i \in S \setminus \{0, N\}$ en $t \in \mathbb{N}$:

$$y_i^t := \max_{a \in A(i)} \{p_{i,i-a}(a)y_{i-a}^{t-1} + p_{i,i+a}(a)y_{i+a}^{t-1}\} = \max_{a \in A(i)} \{(1-p)y_{i-a}^{t-1} + py_{i+a}^{t-1}\}.$$

Verder geldt $y_0^t = y_N^t = 0$ voor alle $t \in \mathbb{N}$. De definitie geeft immers:

$$y_0^t = \max_{a \in A(0)} \sum_j p_{0j}(a)y_j^{t-1} = \sum_j p_{0j}(0)y_j^{t-1}$$

en

$$y_N^t = \max_{a \in A(N)} \sum_j p_{Nj}(a)y_j^{t-1} = \sum_j p_{Nj}(0)y_j^{t-1}.$$

We weten ook dat er geldt $p_{ij}(0) = 0$ voor alle $j \in S$ en $i \in \{0, N\}$. Hieruit volgt onmiddellijk $y_0^t = y_N^t = 0$.

Lemma 4.2

Er geldt:

$$y_i^1 = \begin{cases} 0 & \text{als } i \in \{0, N\} \\ 1 & \text{als } i \in \{1, 2, \dots, N-1\} \end{cases}$$

Bewijs

De stelling is juist voor het geval $i \in \{0, N\}$, want we hebben al eerder gezien dat $y_0^t = y_N^t = 0$ geldt voor alle $t \in \mathbb{N}$. En in het geval $i \in \{1, 2, \dots, N-1\}$ geeft de definitie:

$$y_i^1 = \max_{a \in A(i)} \{(1-p)y_{i-a}^0 + py_{i+a}^0\} = \max_{a \in A(i)} \{(1-p) \cdot 1 + p \cdot 1\} = 1.$$

□

Lemma 4.3

Voor $t \geq 2$ geldt: als $y_0^{t-1}, y_1^{t-1}, \dots, y_{j-1}^{t-1}$ strikt kleiner dan 1 zijn, dan is y_j^t ook strikt kleiner dan 1.

Bewijs

Er geldt

$$y_j^t = \max_{a \in A(j)} \{(1-p)y_{j-a}^{t-1} + py_{j+a}^{t-1}\}.$$

Omdat voor alle $a \in A(j)$ geldt $j-a \in \{0, 1, \dots, j-1\}$, is y_{j-a}^{t-1} strikt kleiner dan 1. Hieruit volgt

$$y_j^t = \max_{a \in A(j)} \{(1-p)y_{j-a}^{t-1} + py_{j+a}^{t-1}\} < \max_{a \in A(j)} \{(1-p) + py_{j+a}^{t-1}\} \leq \max_{a \in A(j)} \{(1-p) + p\} = 1.$$

□

Stelling 4.2

Er geldt $\max_i y_i^{N+1} < 1$ voor alle $i \in S$.

Bewijs

Het is voldoende om te bewijzen dat $y_0^t, y_1^t, \dots, y_{t-1}^t$ alle strikt kleiner dan 1 zijn voor $t \in \mathbb{N}$. Want dan geldt $y_i^{N+1} < 1$ voor alle $i \in S$, dus $\max_i y_i^{N+1} < 1$.

We zullen het bewijzen met volledige inductie naar t . Voor $t = 1$ klopt het, want $y_0^1 = 0$. Stel het juist voor alle natuurlijke getallen kleiner dan t . Dan geldt $y_0^{t-1}, y_1^{t-1}, \dots, y_{t-2}^{t-1}$ zijn allen strikt kleiner dan 1. Nu volgt uit lemma 4.3 onmiddellijk dat $y_1^t, y_2^t, \dots, y_{t-1}^t$ alle strikt kleiner zijn dan 1. Verder weten we al dat $y_0^t = 0 < 1$ geldt. Dus het klopt inderdaad dat $y_0^t, y_1^t, \dots, y_{t-1}^t$ alle strikt kleiner dan 1 zijn. Dus de stelling is juist.

□

We kunnen nu dus concluderen dat het casinomodel een transiënt model is. Het gevolg is dat de resultaten van het verdisconteerde model (het begrip verdisconteerd wordt in het hoofdstuk 5 besproken) ook geldig zijn. Dit volgt uit de volgende twee stellingen.

Stelling 4.3

Het Markov beslissingsmodel is transiënt \Leftrightarrow het model is contraherend.

Bewijs

Zie pagina 42 van L.C.M. Kallenberg, 1983. *Linear programming and finite Markovian control problems*. Amsterdam: Mathematisch Centrum.

□

Het begrip contractie wordt overigens nog in hoofdstuk 6 besproken.

Stelling 4.4

Model is contraherend \Leftrightarrow model is equivalent met een verdisconteerd model.

Bewijs

Zie J.A.E.E. van Nunen, 1976. *Contracting Markov decision processes*. Amsterdam: Mathematisch Centrum.

□

5 Verdisconteerd model

Stel r is het rentepercentage per tijdseenheid. Een bedrag B op tijdstip 0 is dan na t tijdseenheden $(1+r)^t \cdot B$ waard. Verdiscontering wordt gebruikt om opbrengsten in de toekomst in het heden te waarderen. Dit gebeurt door het bedrag dat wordt verkregen op tijdstip t te vermenigvuldigen met $(1+r)^{-t}$. De verdisconteringsfactor α wordt dan als volgt gedefinieerd: $\alpha = (1+r)^{-1}$. Er geldt dus $0 < \alpha \leq 1$. Het nut van verdiscontering is om opbrengsten over een oneindige periode op te kunnen tellen (d.w.z. de som moet eindig zijn). Stel $|r_i(a)| \leq M$ voor alle $i \in S$ en $a \in A(i)$, dan is de verdisconteerde opbrengst over de oneindige horizon eindig, namelijk begrensd door $\sum_{t=0}^{\infty} \alpha^t M = (1-\alpha)^{-1} M < \infty$. De verwachte verdisconteerde opbrengst $v_i^\alpha(R)$, voor een strategie R en begintoestand i , is als volgt gedefinieerd:

$$v_i^\alpha(R) = \sum_{t=1}^{\infty} \alpha^{t-1} \mathbb{E}_{i,R}[r_{X_t}(Y_t)] = \sum_{t=1}^{\infty} \alpha^{t-1} \sum_{j,a} \mathbb{P}_{i,R}[X_t = j, Y_t = a] \cdot r_j(a).$$

Hierbij is $\mathbb{E}_{i,R}$ de verwachtingsoperator met betrekking tot de kansmaat $\mathbb{P}_{i,R}$, X_t de stochastische variabele voor de toestand op tijdstip t , en is Y_t de stochastische variabele voor de actie op tijdstip t . De waarde-vector v_i^α is gedefinieerd als:

$$v_i^\alpha = \max_R v_i^\alpha(R), i \in S.$$

Een strategie R_* wordt een optimale strategie genoemd als geldt: $v_i^\alpha(R_*) = v_i^\alpha$ voor alle $i \in S$.

Voor een stationaire strategie f^∞ is de verwachte verdisconteerde opbrengst $v_i^\alpha(f^\infty)$ (als vector) eenvoudig uit te drukken in de overgangsmatrix $P(f)$ en opbrengst-vector $r(f)$:

$$v_i^\alpha(f^\infty) = \sum_{t=0}^{\infty} \alpha^t P^t(f) r(f) = \left[\sum_{t=0}^{\infty} \alpha^t P^t(f) \right] r(f) = [I - \alpha P(f)]^{-1} r(f).$$

Stelling 5.1

De waarde-vector v^α is de unieke oplossing van de volgende optimaliteitsvergelijking:

$$x_i = \max_{a \in A(i)} \{r_i(a) + \alpha \sum_j p_{ij}(a) x_j\}, i \in S$$

Bewijs

Zie pagina 151 van M.L. Puterman, 1994. *Markov Decision Processes*. New Jersey: Wiley.

□

Stelling 5.2

$v_i^\alpha(f^\infty)$ is de unieke oplossing van

$$x_i = r_i(f) + \alpha \sum_j p_{ij}(f) x_j, i \in S$$

Bewijs

Zie pagina 145 van M.L. Puterman, 1994. *Markov Decision Processes*. New Jersey: Wiley.

□

Stelling 5.3

Als f^∞ voldoet aan

$$v_i^\alpha(f^\infty) \geq r_i(a) + \alpha \sum_j p_{ij}(f) v_j^\alpha(f^\infty) \text{ voor alle } i \in S, a \in A(i),$$

dan is f^∞ optimaal.

Bewijs

Zie pagina 176 van M.L. Puterman, 1994. *Markov Decision Processes*. New Jersey: Wiley.

□

We passen nu de theorie toe op het casinomodel. In het vorige hoofdstuk hebben we geconcludeerd dat dit model transiënt is, dus de resultaten van het verdisconteerde model kunnen worden toegepast met $\alpha = 1$. De verwachte verdisconteerde opbrengst $v_i^\alpha(R)$ kan nu (door de handige keuze van de directe opbrengst) geïnterpreteerd worden als de kans dat het systeem eindigt op toestand N gegeven dat i de begintoestand is. Dit is als volgt in te zien. Omdat $r_N(0) = 1$ en alle overige directe opbrengsten $r_i(a) = 0$, met $i \in S \setminus \{N\}$ en $a \in A(i)$, gelijk zijn aan 0 geldt er:

$$v_i^\alpha(R) = \sum_{t=1}^{\infty} \alpha^{t-1} \sum_{j,a} \mathbb{P}_{i,R}[X_t = j, Y_t = a] \cdot r_j(a) = \sum_{t=1}^{\infty} \mathbb{P}_{i,R}[X_t = N, Y_t = 0].$$

Nu is $\mathbb{P}_{i,R}[X_t = N, Y_t = 0]$ gelijk aan de kans dat de speler na t spellen zijn doel heeft bereikt, dus $\sum_{t=1}^{\infty} \mathbb{P}_{i,R}[X_t = N, Y_t = 0]$ is gelijk aan de kans dat het systeem ooit in toestand N terechtkomt en als N bereikt wordt, stopt het model.

De optimaliteitsvergelijking (met als unieke oplossing de waarde-vector v) voor het casinomodel is:

$$\begin{cases} x_0 = 0 \\ x_i = \max_{a \in A(i)} \{p x_{i+a} + (1-p)x_{i-a}\}, 1 \leq i \leq N-1 \\ x_N = 1 \end{cases}$$

Laat f_1 de strategie zijn waarbij iedere keer 1 euro wordt ingezet.

Stelling 5.4

$$p = \frac{1}{2} \Rightarrow v_i(f_1^\infty) = \frac{i}{N}, 0 \leq i \leq N.$$

Bewijs

We maken gebruik van het feit dat $v_i(f_1^\infty)$ gelijk is aan de kans dat je eindigt in toestand N als je in toestand i bent en iedere keer 1 euro inzet. Nu is onmiddellijk duidelijk dat er geldt $v_0(f_1^\infty) = 0$ en $v_N(f_1^\infty) = 1$. Dus voor $i = 0$ en $i = N$ klopt de stelling.

Voor de overige $i \in S$ is het wat moeilijker in te zien wat de kans is dat je in toestand N eindigt, maar we kunnen wel makkelijk de kansen in elkaar uitdrukken en dan dit stelsel vergelijkingen oplossen. Stel het systeem bevindt zich in toestand i ongelijk aan 0 of N . De kans dat het systeem op het volgende tijdstip in $i - 1$ is gelijk aan $\frac{1}{2}$, en de kans dat het systeem op het volgende tijdstip in $i + 1$ is ook $\frac{1}{2}$. We vinden nu dus de volgende relatie:

$$v_i(f_1^\infty) = \frac{1}{2}v_{i-1}(f_1^\infty) + \frac{1}{2}v_{i+1}(f_1^\infty).$$

Nu hoeven we alleen nog maar te controleren of $v_i(f_1^\infty) = \frac{i}{N}$ voldoet:

$$\frac{1}{2}v_{i-1}(f_1^\infty) + \frac{1}{2}v_{i+1}(f_1^\infty) = \frac{1}{2} \cdot \frac{i-1}{N} + \frac{1}{2} \cdot \frac{i+1}{N} = \frac{i}{N}.$$

Dus $v_i(f_1^\infty) = \frac{i}{N}$ voldoet inderdaad. Dat deze oplossing uniek is, volgt uit stelling 5.2 en hiermee is ook de stelling bewezen. □

Stelling 5.5

$p \neq \frac{1}{2} \Rightarrow v_i(f_1^\infty) = \frac{1-r^i}{1-r^N}$, $0 \leq i \leq N$, waarbij $r = \frac{1-p}{p}$.

Bewijs

Het bewijs gaat op dezelfde wijze als het bewijs in de vorige stelling. Het is duidelijk dat er geldt $v_0(f_1^\infty) = 0$ en $v_N(f_1^\infty) = 1$. Dus voor $i = 0$ en $i = N$ klopt de stelling. Voor de overige 'kansen' $v_i(f_1^\infty)$ geldt de volgende relatie:

$$v_i(f_1^\infty) = (1-p)v_{i-1}(f_1^\infty) + pv_{i+1}(f_1^\infty).$$

Nu hoeven we alleen nog maar te controleren of $v_i(f_1^\infty) = \frac{1-r^i}{1-r^N}$ voldoet:

$$(1-p)v_{i-1}(f_1^\infty) + pv_{i+1}(f_1^\infty) = (1-p) \cdot \frac{1-r^{i-1}}{1-r^N} + p \cdot \frac{1-r^{i+1}}{1-r^N}.$$

De vraag is dus of $1-r^i$ gelijk is aan $(1-p)(1-r^{i-1}) + p(1-r^{i+1})$. En dit kunnen we weer reduceren tot de vraag of r^i gelijk is aan $(1-p)r^{i-1} + pr^{i+1}$. Deze laatste uitdrukking werken we even uit:

$$\begin{aligned} (1-p)r^{i-1} + pr^{i+1} &= \frac{(1-p)^i}{p^{i-1}} + \frac{(1-p)^{i+1}}{p^i} = \frac{p(1-p)^i}{p^i} + \frac{(1-p)^{i+1}}{p^i} \\ &= \frac{p \cdot (1-p)^i + (1-p)(1-p)^i}{p^i} = \frac{(1-p)^i}{p^i} = r^i. \end{aligned}$$

We zien dus dat het klopt. Dat deze oplossing uniek is, volgt uit stelling 5.2 en hiermee is ook de stelling bewezen.

□

Stelling 5.6

Als $p = \frac{1}{2}$, dan is iedere strategie optimaal.

Bewijs

Zij f^∞ een willekeurig stationaire strategie. Dan gelden de volgende relaties:

$$v_i(f^\infty) = \frac{1}{2}v_{i+f(i)}(f^\infty) + \frac{1}{2}v_{i-f(i)}(f^\infty), 1 \leq i \leq N-1, v_0(f^\infty) = 0 \text{ en } v_N(f^\infty) = 1.$$

Merk op dat deze relatie gelijk is aan het stelsel vergelijkingen in stelling 5.2. Nu blijkt $v_i(f^\infty) = \frac{i}{N}$, $1 \leq i \leq N-1$ de (unieke) oplossing te zijn. Dit is eenvoudig te controleren: $\frac{1}{2}v_{i+f(i)}(f^\infty) + \frac{1}{2}v_{i-f(i)}(f^\infty) = \frac{1}{2} \cdot \frac{i+f(i)}{N} + \frac{1}{2} \cdot \frac{i-f(i)}{N} = \frac{i}{N}$. Omdat $\frac{i}{N}$ onafhankelijk is van f^∞ is iedere strategie optimaal.

□

Stelling 5.7

Als $p > \frac{1}{2}$, dan is de strategie f_1^∞ , waarbij iedere keer 1 euro wordt ingezet, optimaal.

Bewijs

Vanwege stelling 5.3 is het voldoende om aan te tonen dat voor alle $i \in S$ en $a \in A(i)$ geldt:

$$v_i(f_1^\infty) \geq r_i(a) + \sum_j p_{ij}(a)v_j(f_1^\infty) = pv_{i+a}(f_1^\infty) + (1-p)v_{i-a}(f_1^\infty).$$

Er geldt $v_i(f_1^\infty) = \frac{1-r^i}{1-r^N}$, $0 \leq i \leq N$ met $r = \frac{1-p}{p}$ (merk op dat er geldt $0 < r < 1$). Laat $q = 1-p$ (er geldt dus $0 < q < \frac{1}{2}$ en $r = \frac{q}{p}$). We moeten dus bewijzen:

$$\frac{1-r^i}{1-r^N} \geq p \cdot \frac{1-r^{i+a}}{1-r^N} + q \cdot \frac{1-r^{i-a}}{1-r^N}.$$

Dit is equivalent met bewijzen dat er geldt

$$1-r^i \geq p(1-r^{i+a}) + q(1-r^{i-a}).$$

Dit is te reduceren tot:

$$-r^i \geq -pr^{i+a} - qr^{i-a}, \text{ m.a.w } 1 \leq pr^a + qr^{-a}$$

Laat $F(a) = pr^a + qr^{-a}$. We bewijzen de bewering $1 \leq F(a)$ met volledige inductie naar a . Stel $a = 1$, dan

$$F(a) = F(1) = pr^1 + qr^{-1} = p \cdot \frac{q}{p} + q \cdot \frac{p}{q} = q + p = 1.$$

Het is nu voldoende om te bewijzen dat er geldt $F(a+1) \geq F(a)$ voor alle a . Dus we moeten bewijzen

$$pr^{a+1} + qr^{-a-1} \geq pr^a + qr^{-a}.$$

Dit komt overeen met

$$pr^{2a+2} + q \geq pr^{2a+1} + qr,$$

en dit is weer hetzelfde als

$$pr^{2a+1}(r - 1) \geq q(r - 1).$$

M.a.w. geldt er $r^{2a+1} \leq r$? Dit is alleen waar als $r \leq 1$, en is dit geval is dat ook zo omdat $p \geq \frac{1}{2}$. Dus hiermee is de stelling bewezen.

□

6 Monotone contractie afbeeldingen

Zij X een reële lineair vectorruimte. X is een genormeerde lineaire ruimte als er een niet-negatieve functie $\|x\|$, genaamd de norm van x , bestaat zodat:

1. $\|x\| = 0 \Leftrightarrow x = 0$;
2. $\|x + y\| \leq \|x\| + \|y\|$ voor alle $x, y \in X$;
3. $\|\beta x\| = |\beta| \cdot \|x\|$ voor alle $x \in X$ en $\beta \in \mathbb{R}$.

Een genormeerde lineaire ruimte X heet een Banach ruimte als voor iedere rij $x^1, x^2, x^3, \dots \in X$ met

$$\lim_{n, m \rightarrow \infty} \|x^n - x^m\| = 0,$$

er een $x^* \in X$ is zodat

$$\lim_{n \rightarrow \infty} \|x^n - x^*\| = 0.$$

Zij X een genormeerde lineaire ruimte en $B : X \rightarrow X$ een afbeelding van X naar X . B heet een contractie afbeelding als er een $\beta \in [0, 1)$ is zodat

$$\|Bx - By\| \leq \beta \|x - y\| \text{ voor alle } x, y \in X.$$

Verder wordt β dan de contractiefactor van B genoemd. En $x^* \in X$ is een vaste punt van B als $Bx^* = x^*$.

Stelling 6.1

Zij X een Banach ruimte en $B : X \rightarrow X$ een contractie afbeelding. Dan:

1. $x^* = \lim_{n \rightarrow \infty} B^n x$ bestaat voor alle $x \in X$, en x^* is een vast punt van B ;
2. x^* is het unieke vaste punt van B .

Bewijs

Zie pagina 251 van J. Stoer en R. Bulirsch, 1980. *Introduction to Numerical Analysis*. Springer.

□

Een partiële ordening op een verzameling X is een relatie \leq op X die aan de volgende eigenschappen voldoet:

1. $x \leq x$ voor alle $x \in X$;
2. $x \leq y \wedge y \leq x \Rightarrow x = y$ voor alle $x, y \in X$;
3. $x \leq y \wedge y \leq z \Rightarrow x \leq z$ voor alle $x, y, z \in X$.

Zij X een partiële ordening en $B : X \rightarrow X$ een afbeelding van X naar X . B heet een monotone afbeelding als $x \leq y$ impliceert dat $Bx \leq By$ geldt. (Als we $x \geq y$ als volgt definiëren: $x \geq y \Leftrightarrow y \leq x$, dan is het triviaal dat er ook geldt $x \geq y \Rightarrow Bx \geq By$.)

Stelling 6.2

Zij X een partieel geordende Banach ruimte en $B : X \rightarrow X$ een contractie afbeelding. Dan geldt:

1. $Bx \leq x \Rightarrow x^* \leq Bx \leq x$;
2. $Bx \geq x \Rightarrow x^* \geq Bx \geq x$.

Bewijs

1. Stel gegeven is $x \geq Bx$. Eerst tonen we aan dat $B^{n-1}x \geq B^n x$ geldt voor alle $n \in \mathbb{N}_0$. Dit gaat met volledige inductie naar n . Voor $n = 0$ klopt het, want $x \geq Bx$ is gegeven. Stel de bewering is waar voor alle natuurlijke getallen kleiner dan n , dan geldt $B^{n-2}x \geq B^{n-1}x$. Maar nu volgt uit de monotonie van B onmiddellijk $BB^{n-2}x \geq BB^{n-1}x$, dus $B^{n-1}x \geq B^n x$. We concluderen nu dat de bewering $B^{n-1}x \geq B^n x$ voor alle $n \in \mathbb{N}_0$ waar is. Hieruit volgt ook

$$x \geq Bx \geq \dots \geq B^n x, n \in \mathbb{N}_0.$$

We kunnen nu eenvoudig het bewijs afmaken:

$$x^* = \lim_{n \rightarrow \infty} B^n x \leq Bx \leq x.$$

2. Het bewijs gaat op dezelfde manier als onderdeel 1., alleen moeten de ongelijkheidstekens omgedraaid worden. Dus \geq moet iedere keer vervangen worden door \leq en andersom.

□

De supremumnorm $\|x\|_\infty$ op een vector $x \in \mathbb{R}^N$ is gedefinieerd als

$$\|x\|_\infty = \max_{1 \leq i \leq N} |x_i|.$$

Het is eenvoudig in te zien (door de drie eigenschappen van een norm na te gaan) dat de supremumnorm een goed gedefinieerde norm is. Merk op dat de volgende ongelijkheid altijd geldt: $x \leq \|x\|_\infty \cdot e$ voor alle $x \in \mathbb{R}^N$, met $e \in \mathbb{R}^N$ de vector met alle elementen gelijk aan 1. Verder is de matrix norm $\|A\|_\infty$ voor een vierkante $N \times N$ matrix A gedefinieerd als:

$$\|A\|_\infty = \max_i \sum_{j=1}^N |a_{ij}|.$$

Met deze definitie is het eenvoudig in te zien dat voor een Markov keten P geldt:

$$\|P\|_\infty = \max_i \sum_j p_{ij} = 1.$$

De \mathbb{R}^N met de supremumnorm en ordening $x \leq y \Leftrightarrow x_i \leq y_i$ voor alle $1 \leq i \leq N$, is een partieel geordende Banach ruimte.

Stelling 6.3

1. Zij $B : \mathbb{R}^N \rightarrow \mathbb{R}^N$ een monotone contractie afbeelding met contractiefactor β en zij $d \in \mathbb{R}$ een scalar. Dan geldt $x \leq y + d \cdot e \Rightarrow Bx \leq By + \beta \cdot |d| \cdot e$;
2. Zij $B : \mathbb{R}^N \rightarrow \mathbb{R}^N$ een afbeelding met de eigenschap $x \leq y + d \cdot e \Rightarrow Bx \leq By + \beta \cdot |d| \cdot e$ voor een $\beta \in [0, 1)$ en voor alle $d \in \mathbb{R}$. Dan is B een monotone contractie afbeelding, t.o.v de supremumnorm, met contractiefactor β .

Bewijs

1. Omdat $x \leq y + d \cdot e$ geldt en B monotoon is volgt hieruit: $Bx \leq B(y + d \cdot e)$. Deze laatste uitdrukking kunnen we nog verder afschatten:

$$\begin{aligned} B(y + d \cdot e) &= B(y + d \cdot e) - By + By \leq \|B(y + d \cdot e) - By\|_\infty \cdot e + By \\ &\leq \beta \cdot \|(y + d \cdot e) - y\|_\infty \cdot e + By = \beta \cdot |d| \cdot e + By \end{aligned}$$

2. De eigenschap $x \leq y + d \cdot e \Rightarrow Bx \leq By + \beta \cdot |d| \cdot e$ geldt voor alle $d \in \mathbb{R}$, dus in het bijzonder voor $d = 0$. Als $d = 0$ dan staat er $x \leq y \Rightarrow Bx \leq By$. Hieruit volgt dus dat B monotoon is. We weten dat de ongelijkheid $x - y \leq \|x - y\|_\infty \cdot e$ geldt. Passen we nu de eigenschap van B toe met $d = \|x - y\|_\infty$, dan krijgen we het volgende resultaat:

$$Bx - By \leq \beta \cdot \|x - y\|_\infty \cdot e.$$

We weten ook dat $y - x \leq \|x - y\|_\infty \cdot e$ geldt. En als we nu de eigenschap van B gebruiken krijgen we

$$By - Bx \leq \beta \cdot \|x - y\|_\infty \cdot e.$$

Uit de laatste twee verkregen ongelijkheden volgt nu

$$\|Bx - By\|_\infty \leq \beta \cdot \|x - y\|_\infty.$$

Dus B is een contractie afbeelding met contractiefactor β .

□

Lemma 6.1

Zij $B : \mathbb{R}^N \rightarrow \mathbb{R}^N$ een monotone contractie afbeelding, t.o.v de supremumnorm, met contractiefactor β en vaste punt x^* . Stel er bestaan scalars a en b zodat $a \cdot e \leq Bx - x \leq b \cdot e$ voor een $x \in \mathbb{R}^N$. Dan geldt

$$x - (1 - \beta)^{-1} |a| \cdot e \leq Bx - \beta(1 - \beta)^{-1} |a| \cdot e \leq x^* \leq Bx + \beta(1 - \beta)^{-1} |b| \cdot e \leq x + (1 - \beta)^{-1} |b| \cdot e.$$

Bewijs

Er geldt $Bx \leq x + b \cdot e \leq x + |b| \cdot e$. Omdat B monotoon is geldt nu ook

$$\begin{aligned} B^2x &\leq B(x + |b| \cdot e) = B(x + |b| \cdot e) - Bx + Bx \leq \|B(x + |b| \cdot e) - Bx\|_\infty \cdot e + Bx \\ &\leq Bx + \beta|b| \cdot e \leq x + (1 + \beta)|b| \cdot e. \end{aligned}$$

Als we deze stap herhaaldelijk toepassen, is het m.b.v inductie naar n aan te tonen dat de volgende ongelijkheid geldt:

$$B^n x \leq Bx + (\beta + \dots + \beta^{n-1})|b| \cdot e \leq x + (1 + \beta + \dots + \beta^{n-1})|b| \cdot e, n \in \mathbb{N}.$$

We kunnen nu een bovengrens vinden voor x^* :

$$\begin{aligned} x^* &= \lim_{n \rightarrow \infty} B^n x \leq \lim_{n \rightarrow \infty} (Bx + (\beta + \dots + \beta^{n-1})|b| \cdot e) = Bx + \beta(1 - \beta)^{-1}|b| \cdot e \\ &\leq \lim_{n \rightarrow \infty} (x + (1 + \beta + \dots + \beta^{n-1})|b| \cdot e) = x + (1 - \beta)^{-1}|b| \cdot e. \end{aligned}$$

Door met de ongelijkheid $Bx \geq x + a \cdot e$ te beginnen, kunnen we op soortgelijke wijze een ondergrens vinden voor x^* :

$$x^* \geq Bx - \beta(1 - \beta)^{-1}|a| \cdot e \geq x - (1 - \beta)^{-1}|a| \cdot e.$$

□

Een gevolg van deze stelling is dat voor een monotone contractie afbeelding B , t.o.v. de supremumnorm, met contractiefactor β en vaste punt x^* geldt:

$$\begin{aligned} x - (1 - \beta)^{-1}\|Bx - x\|_\infty \cdot e &\leq Bx - \beta(1 - \beta)^{-1}\|Bx - x\|_\infty \cdot e \\ &\leq x^* \leq Bx + \beta(1 - \beta)^{-1}\|Bx - x\|_\infty \cdot e \leq x + (1 - \beta)^{-1}\|Bx - x\|_\infty \cdot e. \end{aligned}$$

Dit volgt door lemma 6.1 toe te passen op de ongelijkheid:

$$-\|Bx - x\|_\infty \cdot e \leq Bx - x \leq \|Bx - x\|_\infty \cdot e.$$

Lemma 6.2

Zij $B : \mathbb{R}^N \rightarrow \mathbb{R}^N$ een monotone contractie afbeelding met contractiefactor β , vaste punt x^* en met de eigenschap $B(x + c \cdot e) = Bx + Bc \cdot e$ voor alle $x \in \mathbb{R}^N$ en c een scalar. Stel er bestaan scalars a en b zodat $a \cdot e \leq Bx - x \leq b \cdot e$ voor een $x \in \mathbb{R}^N$. Dan geldt:

$$x - (1 - \beta)^{-1}a \cdot e \leq Bx - \beta(1 - \beta)^{-1}a \cdot e \leq x^* \leq Bx + \beta(1 - \beta)^{-1}b \cdot e \leq x + (1 - \beta)^{-1}b \cdot e.$$

Bewijs

B is monotoon en $Bx \leq x + b \cdot e$ geldt, dus hieruit volgt:

$$B^2x \leq B(x + b \cdot e) = Bx + \beta \cdot e \leq x + (1 + \beta)b \cdot e.$$

Verder is met inductie naar n het volgende aan te tonen:

$$B^n x \leq Bx + (\beta + \beta^2 + \dots + \beta^{n-1})b \cdot e \leq x + (1 + \beta + \beta^2 + \dots + \beta^{n-1})b \cdot e.$$

Nu is eenvoudig een bovengrens voor x^* te bepalen:

$$\begin{aligned}x^* &= \lim_{n \rightarrow \infty} B^n x \leq \lim_{n \rightarrow \infty} (Bx + (\beta + \beta^2 + \dots + \beta^{n-1})b \cdot e) = Bx + \beta(1 - \beta)^{-1}b \cdot e \\ &\leq \lim_{n \rightarrow \infty} (x + (1 + \beta + \beta^2 + \dots + \beta^{n-1})b \cdot e) = x + (1 - \beta)^{-1}b \cdot e.\end{aligned}$$

Op een soortgelijke wijze is de ondergrens voor x^* te vinden door met de ongelijkheid $Bx \geq x + a \cdot e$ te beginnen.

□

7 Optimaliteitsvergelijking

We introduceren eerst twee afbeeldingen. Definieer de afbeelding $U : \mathbb{R}^N \rightarrow \mathbb{R}^N$ als volgt:

$$(Ux)_i = \max_{a \in A(i)} \{r_i(a) + \alpha \sum_j p_{ij}(a)x_j\}, i \in S.$$

Dan is v^α een vaste punt van de afbeelding U , $Ux = x$ is immers gelijk aan de optimaliteitsvergelijking uit hoofdstuk 5 (zie stelling 5.1). Verder definiëren we voor ieder willekeurige beslisregel b de afbeelding $L_b : \mathbb{R}^N \rightarrow \mathbb{R}^N$ als volgt:

$$L_b x = r(b) + \alpha P(b)x.$$

Laat $f_x(i)$ voldoen aan:

$$r_i(f_x(i)) + \alpha \sum_j p_{ij}(f_x(i))x_j = \max_a \{r_i(a) + \alpha \sum_j p_{ij}(a)x_j\}, i \in S,$$

dan geldt:

$$L_{f_x} x = Ux = \max_f L_f x.$$

Stelling 7.1

L_b en U zijn t.o.v. de supremumnorm monotone contractie afbeeldingen in \mathbb{R}^N met contractiefactor α .

Bewijs

Stel $x \geq y$ en b een willekeurig stationaire beslisregel. Als L_b monotoon is, moet er gelden $L_b x \geq L_b y$. Dit is zo, want (merk op dat $P(b) \geq 0$ geldt):

$$L_b x = r(b) + \alpha P(b)x \geq r(b) + \alpha P(b)y = L_b y.$$

Nu is ook eenvoudig aan te tonen dat U monotoon is:

$$Ux = \max_f L_f x \geq L_{f_y} x \geq L_{f_y} y = Uy.$$

Verder geldt,

$$L_b x - L_b y = \{r(b) + \alpha P(b)x\} - \{r(b) + \alpha P(b)y\} = \alpha P(b)(x - y),$$

dus dit geeft

$$\|L_b x - L_b y\|_\infty = \|\alpha P(b)(x - y)\|_\infty \leq \alpha \cdot \|P(b)\|_\infty \cdot \|x - y\|_\infty = \alpha \cdot \|x - y\|_\infty.$$

Hieruit volgt dat L_b een contractie afbeelding is met contractiefactor α . Nu moeten we nog bewijzen dat U ook een contractie afbeelding is met contractiefactor α :

$$\begin{aligned} Ux - Uy &= L_{f_x} x - L_{f_y} y \leq L_{f_x} x - L_{f_x} y = \{r(f_x) + \alpha P(f_x)x\} - \{r(f_x) + \alpha P(f_x)y\} \\ &= \alpha \cdot P(f_x)(x - y) \leq \alpha \cdot \|P(f_x)\|_\infty \cdot \|x - y\|_\infty \cdot e = \alpha \cdot \|x - y\|_\infty \cdot e. \end{aligned}$$

Er geldt $Ux \geq Uy$, dus

$$Uy - Ux \leq Ux - Uy \leq \alpha \cdot \|x - y\|_\infty \cdot e.$$

Uit de ongelijkheden $Ux - Uy \leq \alpha \cdot \|x - y\|_\infty \cdot e$ en $Uy - Ux \leq \alpha \cdot \|x - y\|_\infty \cdot e$ volgt nu

$$\|Ux - Uy\|_\infty \leq \alpha \cdot \|x - y\|_\infty.$$

Dus ook U is een contractie afbeelding is met contractiefactor α .

□

Stelling 7.2

$v^\alpha(b^\infty)$ is het vaste punt van de afbeelding L_b , m.a.w. $v^\alpha(b^\infty)$ is de unieke oplossing van de vergelijking $L_b x = x$.

Bewijs

Vanwege stelling 6.1 en stelling 7.1 is het voldoende om aan te tonen dat er geldt:

$$L_b v^\alpha(b^\infty) = v^\alpha(b^\infty), \text{ oftewel } L_b v^\alpha(b^\infty) - v^\alpha(b^\infty) = 0.$$

Dit is als volgt aan te tonen:

$$L_b v^\alpha(b^\infty) - v^\alpha(b^\infty) = r(b) + \alpha P(b) v^\alpha(b^\infty) - v^\alpha(b^\infty) = r(b) - [I - \alpha P(b)] v^\alpha(b^\infty)$$

Omdat $v^\alpha(b^\infty)$ gelijk is aan $[I - \alpha P(b)]^{-1} r(b)$, krijgen we nu:

$$r(b) - [I - \alpha P(b)] v^\alpha(b^\infty) = r(b) - [I - \alpha P(b)] [I - \alpha P(b)]^{-1} r(b) = 0.$$

□

De stellingen 6.1 en 7.2 hebben als gevolg dat voor alle $x \in \mathbb{R}^N$ geldt:

$$v^\alpha(b^\infty) = \lim_{n \rightarrow \infty} L_b^n x.$$

Stelling 7.3

v^α is het vaste punt van de afbeelding U , m.a.w. v^α is de unieke oplossing van de vergelijking $Ux = x$.

Bewijs

Vanwege stelling 6.1 en stelling 7.1 is het voldoende om aan te tonen dat er geldt:

$$Uv^\alpha = v^\alpha.$$

We zullen dit aantonen door te laten zien dat de volgende twee ongelijkheden gelden:

$$Uv^\alpha \leq v^\alpha \text{ en } Uv^\alpha \geq v^\alpha.$$

Zij $R = (b^1, b^2, \dots)$ een willekeurige geheugenloze strategie, dan geldt:

$$v^\alpha = \sum_{t=1}^{\infty} \alpha^{t-1} P(b^1) P(b^2) \dots P(b^{t-1}) r(b^t) = r(b^1) + \sum_{t=2}^{\infty} \alpha^{t-1} P(b^1) P(b^2) \dots P(b^{t-1}) r(b^t)$$

$$= r(b^1) + \alpha P(b^1) \sum_{t=1}^{\infty} \alpha^{t-1} P(b^2) P(b^3) \dots P(b^t) r(b^{t+1}) = r(b^1) + \alpha P(b^1) v^\alpha(R_2) = L_{b^1} v^\alpha(R_2),$$

waarbij $R_2 = (b^2, b^3, \dots)$. Verder geldt

$$v^\alpha(R) = L_{b^1} v^\alpha(R_2) \leq L_{b^1} v^\alpha \leq U v^\alpha,$$

want de eerste ongelijkheid volgt uit de monotonie van L_{b^1} en de tweede ongelijkheid uit de definitie van U . Hieruit volgt

$$v^\alpha = \sup_R v^\alpha(R) \leq U v^\alpha.$$

Nu hoeven we alleen nog de omgekeerde ongelijkheid te bewijzen. Neem $\epsilon > 0$ willekeurig. Omdat $v^\alpha = \sup_R v^\alpha(R)$ geldt, bestaat voor alle $j \in S$ een geheugenloze strategie $R_j^\epsilon = (b^1(j), b^2(j), \dots)$ zodanig dat:

$$v_j^\alpha(R_j^\epsilon) \geq v_j^\alpha - \epsilon.$$

Laat $a_i \in A(i)$ voldoen aan:

$$r_i(a_i) + \alpha \sum_j p_{ij}(a_i) v_j^\alpha = \max_{a \in A(i)} \{r_i(a) + \alpha \sum_j p_{ij}(a) v_j^\alpha\}, i \in S.$$

Definieer nu een strategie $R^* = (b^1, b^2, \dots)$ als volgt:

$$b_{ia}^1 = \begin{cases} 1 & \text{als } a = a_i \\ 0 & \text{als anders} \end{cases} \quad \text{en } b_{i_1 a_1 \dots i_t a}^t = b_{i_t a}^{t-1}(i_2), a \in A(i_t), t \geq 2.$$

Dan geldt

$$\begin{aligned} v_i^\alpha &\geq v_i^\alpha(R^*) = r_i(a_i) + \alpha \sum_j p_{ij}(a_i) v_j^\alpha(R_j^\epsilon) \geq r_i(a_i) + \alpha \sum_j p_{ij}(a_i) v_j^\alpha (v_j^\alpha - \epsilon) \\ &= \max_{a \in A(i)} \{r_i(a) + \alpha \sum_j p_{ij}(a) v_j^\alpha\} = (U v^\alpha)_i - \alpha \epsilon, i \in S. \end{aligned}$$

Omdat ϵ willekeurig gekozen was, volgt hieruit

$$v^\alpha \geq U v^\alpha.$$

□

Uit de stellingen 6.1 en 7.3 volgt nu dat er geldt:

$$v^\alpha = \lim_{n \rightarrow \infty} U^n x \text{ voor alle } x \in \mathbb{R}^N.$$

Lemma 7.1

Voor alle $x \in \mathbb{R}^N$ geldt:

$$1. \quad x - (1 - \alpha)^{-1} \|Ux - x\|_\infty \cdot e \leq Ux - \alpha(1 - \alpha)^{-1} \|Ux - x\|_\infty \cdot e \leq v^\alpha(f_x^\infty) \leq v^\alpha \leq Ux + \alpha(1 - \alpha)^{-1} \|Ux - x\|_\infty \cdot e \leq x + (1 - \alpha)^{-1} \|Ux - x\|_\infty \cdot e$$

$$2. \|v^\alpha - x\|_\infty \leq (1 - \alpha)^{-1} \|Ux - x\|_\infty;$$

$$3. \|v^\alpha - v^\alpha(f_x^\infty)\|_\infty \leq 2\alpha(1 - \alpha)^{-1} \|Ux - x\|_\infty.$$

Bewijs

1. Kies $x \in \mathbb{R}^N$ willekeurig en laat $a = -\|Ux - x\|_\infty$, $b = \|Ux - x\|_\infty$ en $B = L_{f_x}$. Dan volgt uit lemma 6.2 (vervang β door α):

$$x - (1 - \alpha)^{-1} \|Ux - x\|_\infty \cdot e \leq Ux - \alpha(1 - \alpha)^{-1} \|Ux - x\|_\infty \cdot e \leq v^\alpha(f_x^\infty) \leq v^\alpha.$$

Neem nu $B = U$ (i.p.v $B = L_{f_x}$) en pas nu weer lemma 6.2 toe. Nu krijgen we:

$$v^\alpha \leq Ux + \alpha(1 - \alpha)^{-1} \|Ux - x\|_\infty \cdot e \leq x + (1 - \alpha)^{-1} \|Ux - x\|_\infty \cdot e.$$

2. Dit volgt onmiddellijk uit onderdeel 1. van deze stelling.
3. Dit volgt onmiddellijk uit onderdeel 1. van deze stelling.

□

8 Successieve approximatie

Een vector $v \in \mathbb{R}^N$ heet een δ -approximatie van v^α als $\|v^\alpha - v\|_\infty \leq \delta$ geldt voor $\delta > 0$. Een strategie R is een ε -optimale strategie als $\|v^\alpha - v^\alpha(R)\|_\infty \leq \varepsilon$ geldt voor $\varepsilon > 0$. Definieer de rij $\{v^n\}_{n=1}^\infty$ als volgt:

$$\begin{cases} v^1 \in \mathbb{R}^N & \text{willekeurig gekozen} \\ v^{n+1} = Uv^n & \text{als } n = 1, 2, \dots \end{cases}$$

met de bijbehorende rij $f_1^\infty, f_2^\infty, \dots$ van strategieën, waarbij $f_n = f_{v^n}$ voor alle $n \in \mathbb{N}$. Deze rij zal convergeren naar v^α , omdat $v^\alpha = \lim_{n \rightarrow \infty} U^n x$ geldt voor alle $x \in \mathbb{R}^N$. Verder geldt ook:

$$v^{n+1} = Uv^n = L_{f_n} v^n = r(f_n) + \alpha P(f_n) v^n, n \in \mathbb{N}.$$

Stelling 8.1

$\|v^\alpha(f_n^\infty) - v^\alpha\|_\infty \leq 2\alpha^n(1 - \alpha)^{-1} \cdot \|v^2 - v^1\|_\infty$ voor alle $n \in \mathbb{N}$.

Bewijs

Er geldt (de eerste ongelijkheid volgt uit lemma 7.1 onderdeel 3.):

$$\begin{aligned} \|v^\alpha - v^\alpha(f_x^\infty)\|_\infty &\leq 2\alpha(1 - \alpha)^{-1} \|Uv^n - v^n\|_\infty = 2\alpha(1 - \alpha)^{-1} \|Uv^n - Uv^{n-1}\|_\infty \\ &\leq 2\alpha^2(1 - \alpha)^{-1} \|v^n - v^{n-1}\|_\infty = 2\alpha^2(1 - \alpha)^{-1} \|Uv^{n-1} - Uv^{n-2}\|_\infty \\ &\leq \dots \leq 2\alpha^n(1 - \alpha)^{-1} \|v^2 - v^1\|_\infty, n \in \mathbb{N}. \end{aligned}$$

□

Deze stelling houdt dus in dat f_n^∞ een ε -optimale strategie is, als n voldoende groot genomen wordt.

Algoritme voor successieve approximatie:

1. Kies een willekeurige $\varepsilon > 0$ en een willekeurige $x \in \mathbb{R}^N$.
2. (a) Bereken $y_i = \max_{a \in A(i)} \{r_i(a) + \alpha \sum_j p_{ij}(a)x_j\}, i \in S$;
(b) Laat $f(i) = \operatorname{argmax}_{a \in A(i)} \{r_i(a) + \alpha \sum_j p_{ij}(a)x_j\}, i \in S$.
3. Als $\|y - x\|_\infty \leq \frac{1}{2}(1 - \alpha)\alpha^{-1}\varepsilon$, dan is f^∞ een $\frac{1}{2}\varepsilon$ -approximatie van de waardevector v^α (*STOP*);
Anders: $x := y$ en ga naar stap 2.

Stelling 8.2

Het algoritme voor successieve approximatie is correct.

Bewijs

Dat het algoritme eindig is, is makkelijk in te zien. Want $\{U^n x\}_{n=1}^\infty$ convergeert naar v^α , dus $\|y - x\|_\infty$ zal op den duur (na eindig aantal stappen) willekeurig klein worden. En dan stopt ook het algoritme. Stel het algoritme is gestopt met

de bijbehorende verkregen waarden voor x , y en f . Hierbij geldt dus $y = Ux$ en $f = f_x$. Uit (het bewijs van) stelling 8.1 volgt nu

$$\|v^\alpha(f^\infty) - v^\alpha\|_\infty \leq 2\alpha(1 - \alpha)^{-1} \cdot \|y - x\|_\infty \leq \varepsilon,$$

dus f^∞ is ε -optimaal. Verder hebben we

$$\|v^\alpha - y\|_\infty = \|Uv^\alpha - Ux\|_\infty \leq \alpha\|v^\alpha - x\|_\infty \leq \alpha(1 - \alpha)^{-1} \cdot \|y - x\|_\infty \leq \frac{1}{2}\varepsilon,$$

aangezien de tweede ongelijkheid volgt uit onderdeel 2. van lemma 7.1. Dus y is een $\frac{1}{2}\varepsilon$ -approximatie.

□

9 De winkans is kleiner dan een half

In dit hoofdstuk nemen we aan dat voor de winkans p geldt: $p \leq \frac{1}{2}$. We definiëren eerst $v_n(i)$ met $n \in \mathbb{N}_0$ en $i \in \mathbb{Z}$ als volgt:

$$v_0(i) = \begin{cases} 0 & \text{als } i < N \\ 1 & \text{als } i \geq N \end{cases}$$

en

$$v_n(i) = \max_{a \in A(i)} \{pv_{n-1}(i+a) + (1-p)v_{n-1}(i-a)\} \text{ voor } n > 0.$$

Dan is $v_n(i)$ hetzelfde als de kans dat het doel wordt bereikt binnen n spellen als i de begintoestand is en de optimale strategie wordt gebruikt. We spreken verder af dat $v_n(i) = 0$ als $i \leq 0$ en $v_n(i) = 1$ als $i \geq N$. Definieer nu $D_n(s, t)$ voor $0 \leq t \leq s \leq N$, met $s, t \in \mathbb{N}_0$ en $n \in \mathbb{N}_0$, als volgt:

$$D_n(s, t) = pv_n(2s) + (1-p)v_n(2s-N) - pv_n(s+t) - (1-p)v_n(s-t).$$

We zullen nu een intuïtieve betekenis geven aan $D_n(s, t)$. Eerst bekijken we de uitdrukking $pv_n(2s) + (1-p)v_n(2s-N)$. Dit is gelijk aan de kans op succes binnen $n+1$ spellen als je nu in toestand s bent en maximaal (d.w.z. $\min\{i, N-i\}$) inzet. Dit is als volgt in te zien. Stel $s < \frac{N}{2}$ is de huidige toestand en er wordt s ingezet. Met kans p is de volgende toestand dan $2s$ en met kans $1-p$ is 0 de volgende toestand. (Merk op dat in dit geval geldt $v_n(0) = v_n(2s-N) = 0$.) En als $s \geq \frac{N}{2}$ dan wordt er $N-s$ ingezet. Dan is met kans p de volgende toestand N en met kans $1-p$ is de volgende toestand $s - (N-s) = 2s - N$. (In dit geval geldt $v_n(N) = v_n(2s) = 1$.) Verder is de uitdrukking $pv_n(s+t) + (1-p)v_n(s-t)$ gelijk aan de slagingskans binnen $n+1$ spellen als je nu in toestand s bent en je legt $t \leq s$ in. Want als de huidige toestand s is en t ingezet wordt, dan is de volgende toestand $s+t$ met kans p en $s-t$ met kans $1-p$. Dus $D_n(s, t)$ is niets anders dan het verschil tussen de twee genoemde slagingskansen.

In dit hoofdstuk gaan we bewijzen dat als $p \leq \frac{1}{2}$ geldt, dan is bij iedere spel maximaal inzetten een optimale strategie. We zullen dit aantonen door te laten zien dat als de speler maar een eindig aantal spellen kan spelen, dan is maximaal inzetten optimaal. Een spel komt namelijk overeen met een stap uit het *Algoritme voor successieve approximatie* uit hoofdstuk 8. Het is voldoende te bewijzen dat $D_n(s, t) \geq 0$ geldt voor alle $n \in \mathbb{N}_0$ en $0 \leq t \leq s \leq N$. Want $D_n(s, t) \geq 0$ betekent dat maximaal inzetten optimaal is tijdens het komende spel als de speler nog maar $n+1$ spellen mag spelen. Verder zullen we de extra eigenschap dat $v_n(i)$ monotoon stijgend is in i aantonen voor alle $n \in \mathbb{N}_0$. (Dit betekent dat als de speler meer geld in zijn bezit heeft, hij meer kans maakt om zijn doel te bereiken.) Dit alles zullen we bewijzen met volledige inductie naar n . De structuur van dit hoofdstuk is als volgt: stelling 9.1 geeft het bewijs voor $n = 0$ en de stellingen 9.2 en 9.3 vormen tezamen de inductiestap.

Stelling 9.1

$v_0(i)$ is niet-dalend in i , en er geldt $D_0(s, t) \geq 0$ voor $0 \leq t \leq s \leq N$.

Bewijs

Uit de definitie van $v_0(i)$ volgt onmiddellijk dat $v_0(i)$ niet-dalend in i is. Verder hebben we

$$D_0(s, t) = pv_0(2s) + (1-p)v_0(2s-N) - pv_0(s+t) - (1-p)v_0(s-t).$$

We onderscheiden drie gevallen:

- Stel $0 \leq s < \frac{1}{2}N$, dan krijgen we

$$D_0(s, t) = p \cdot 0 + (1-p) \cdot 0 - p \cdot 0 - (1-p) \cdot 0 = 0 \geq 0.$$

- Stel $\frac{1}{2}N \leq s < N$, dan geldt

$$D_0(s, t) = p \cdot 1 + (1-p) \cdot 0 - pv_0(s+t) - (1-p) \cdot 0 = p - pv_0(s+t).$$

Omdat $v_0(s+t)$ gelijk is aan 0 of 1 volgt hieruit dat $D_0(s, t)$ gelijk is aan p of 0. Hieruit volgt $D_0(s, t) \geq 0$.

- Stel $s = N$, dan hebben we

$$D_0(s, t) = p \cdot 1 + (1-p) \cdot 1 - p \cdot 1 - (1-p)v_0(s-t) = (1-p) - (1-p)v_0(s-t).$$

We weten dat $v_0(s-t)$ gelijk is aan 0 of 1, dus $D_0(s, t)$ is gelijk aan $1-p$ of 0. Hieruit volgt $D_0(s, t) \geq 0$.

□

Nu zullen we de inductiestap bewijzen. Dus vanaf nu nemen we aan dat de bewering $D_n(s, t) \geq 0$ geldt voor iedere s, t met $0 \leq t \leq s \leq N$. Merk op dat als $D_n(s, t) \geq 0$ dan geldt in het algemeen de volgende recursie:

$$v_{n+1}(i) = \begin{cases} pv_n(2i) & \text{als } 0 \leq i < \frac{1}{2}N \\ p + (1-p)v_n(2i-N) & \text{als } \frac{1}{2}N \leq i \leq N \end{cases}$$

We nemen ook aan dat $v_n(i)$ monotoon stijgend is in i .

Stelling 9.2

$v_{n+1}(i)$ is monotoon stijgend in i .

Bewijs

We moeten dus bewijzen dat $v_{n+1}(b) \geq v_{n+1}(a)$ geldt als $b > a$. We onderscheiden drie gevallen voor a en b :

- Stel $0 \leq a < b < \frac{N}{2}$. Dan geldt

$$v_{n+1}(a) = pv_n(2a) \text{ en } v_{n+1}(b) = pv_n(2b).$$

We weten dat $v_n(2b) \geq v_n(2a)$ geldt en hieruit volgt meteen $v_{n+1}(b) \geq v_{n+1}(a)$.

- Stel $\frac{N}{2} \leq a < b \leq N$. Dan hebben we:

$$v_{n+1}(a) = p + (1-p)v_n(2a-N) \text{ en } v_{n+1}(b) = p + (1-p)v_n(2b-N).$$

Er geldt $v_n(2b-N) \geq v_n(2a-N)$ en dit geeft weer $v_{n+1}(b) \geq v_{n+1}(a)$.

- Stel $0 \leq a < \frac{N}{2} \leq b \leq N$. Dan krijgen we:

$$v_{n+1}(a) = pv_n(2a) \text{ en } v_{n+1}(b) = p + (1-p)v_n(2b-N).$$

Omdat $p \geq pv_n(2a)$ zien we nu duidelijk dat $v_{n+1}(b) \geq v_{n+1}(a)$ geldt.

□

Nu is het bewijs rond (stellingen 9.1 en 9.2) dat $v_n(i)$ monotoon stijgend is in i voor alle $n \in N_0$.

Lemma 9.1

$v_n(a+b) \geq v_n(a) + v_n(b)$ als $a, b \geq 0$ en $a+b \leq N$.

Bewijs

We bewijzen het met volledige inductie naar n . Vanwege symmetrie kunnen we aannemen dat $a \geq b$ geldt.

Stel $n = 0$. Als $a+b < N$, dan geldt $v_n(a+b) = 0$, maar ook $v_n(a) = v_n(b) = 0$. Dus in dit geval klopt de stelling. Als $a+b = N$, $a < N$ en $b < N$, dan hebben we $v_n(a+b) = 1$ en $v_n(a) = v_n(b) = 0$, dus ook dit geval gaat goed. En als $a = N$ en $b = 0$, dan geldt $v_n(a+b) = 1$, $v_n(a) = 1$ en $v_n(b) = 0$. En de stelling is wederom juist. Dus als $n = 0$, dan is de stelling correct.

Stel de stelling is juist voor alle natuurlijke getallen kleiner dan n . We onderscheiden drie situaties voor a en b :

- Stel $a+b < \frac{N}{2}$. Dan geldt ook $a < \frac{N}{2}$ en $b < \frac{N}{2}$, dus geldt

$$\begin{aligned} v_n(a+b) &= pv_{n-1}(2a+2b) \geq p\{v_{n-1}(2a) + v_{n-1}(2b)\} \\ &= pv_{n-1}(2a) + pv_{n-1}(2b) = v_n(a) + v_n(b). \end{aligned}$$

- Stel $a+b \geq \frac{N}{2}$ en $a < \frac{N}{2}$. Merk op dat dan ook $b < \frac{N}{2}$ geldt. Verder hebben we

$$D_{n-1}(s, t) \geq 0 \text{ voor alle } s, t \text{ met } 0 \leq t \leq s \leq N,$$

dus in het bijzonder voor $s = a+b$ en $t = a-b$. Omdat

$$\begin{aligned} D_{n-1}(a+b, a-b) &= pv_{n-1}(2a+2b) + (1-p)v_{n-1}(2a+2b-N) - pv_{n-1}(2a) - (1-p)v_{n-1}(2b) \\ &= p + (1-p)v_{n-1}(2a+2b-N) - pv_{n-1}(2a) - (1-p)v_{n-1}(2b), \end{aligned}$$

volgt hieruit

$$p + (1-p)v_{n-1}(2a+2b-N) \geq pv_{n-1}(2a) + (1-p)v_{n-1}(2b)$$

$$\geq pv_{n-1}(2a) + pv_{n-1}(2b) = v_n(a) + v_n(b).$$

Er geldt ook

$$v_n(a+b) = p + (1-p)v_{n-1}(2a+2b-N),$$

en dit geeft nu

$$v_n(a+b) \geq v_n(a) + v_n(b).$$

- Stel $a+b \geq \frac{N}{2}$ en $a \geq \frac{N}{2}$. Vanwege $a+b \leq N$ geldt dan $b \leq \frac{N}{2}$. Nu hebben we

$$\begin{aligned} v_n(a+b) &= p + (1-p)v_{n-1}(2a+2b-N) \geq p + (1-p)\{v_{n-1}(2a-N) + v_{n-1}(2b)\} \\ &= \{p + (1-p)v_{n-1}(2a-N)\} + (1-p)v_{n-1}(2b) \\ &\geq \{p + (1-p)v_{n-1}(2a-N)\} + pv_{n-1}(2b) = v_n(a) + v_n(b). \end{aligned}$$

□

Lemma 9.2

$$s \leq \frac{N}{2}, s+t \leq \frac{N}{2} \Rightarrow D_{n+1}(s,t) = pD_n(2s,2t)$$

Bewijs

Er geldt

$$\begin{aligned} D_{n+1}(s,t) &= pv_{n+1}(2s) + (1-p)v_{n+1}(2s-N) - pv_{n+1}(s+t) - (1-p)v_{n+1}(s-t) \\ &= pv_{n+1}(2s) + (1-p) \cdot 0 - p\{pv_n(2s+2t)\} - (1-p)pv_n(2s-2t). \end{aligned}$$

Stel $s < \frac{1}{4}N$ geldt, dan krijgen we

$$\begin{aligned} D_{n+1}(s,t) &= p\{pv_n(4s)\} - p^2v_n(2s+2t) - p(1-p)v_n(2s-2t) \\ &= p^2v_n(4s) - p^2v_n(2s+2t) - p(1-p)v_n(2s-2t). \end{aligned}$$

En als $s \geq \frac{1}{4}N$ geldt dan hebben we:

$$\begin{aligned} D_{n+1}(s,t) &= p\{p + (1-p)v_n(4s-N)\} - p^2v_n(2s+2t) - p(1-p)v_n(2s-2t) \\ &= p^2 + p(1-p)v_n(4s-N) - p^2v_n(2s+2t) - p(1-p)v_n(2s-2t). \end{aligned}$$

Nu werken we het rechterlid van de vergelijking uit:

$$pD_n(2s,2t) = p\{pv_n(4s) + (1-p)v_n(4s-N) - pv_n(2s+2t) - (1-p)v_n(2s-2t)\}$$

Stel $s < \frac{1}{4}N$ geldt, dit geeft

$$\begin{aligned} pD_n(2s,2t) &= p\{pv_n(4s) + (1-p) \cdot 0 - pv_n(2s+2t) - (1-p)v_n(2s-2t)\} \\ &= p^2v_n(4s) - p^2v_n(2s+2t) - p(1-p)v_n(2s-2t). \end{aligned}$$

En als $s \geq \frac{1}{4}N$ geldt dan krijgen we:

$$\begin{aligned} pD_n(2s,2t) &= p\{p \cdot 1 + (1-p)v_n(4s-N) - pv_n(2s+2t) - (1-p)v_n(2s-2t)\} \\ &= p^2 + p(1-p)v_n(4s-N) - p^2v_n(2s+2t) - p(1-p)v_n(2s-2t). \end{aligned}$$

□

Lemma 9.3

$$s \leq \frac{N}{2}, s+t \geq \frac{N}{2} \Rightarrow D_{n+1}(s, t) = p(1-p)\{v_n(4s-N) - v_n(2s+2t-N) - v_n(2s-2t)\}$$

Bewijs

Merk op dat nu ook geldt $s-t \leq \frac{N}{2}$ en $2s \geq s+t \geq \frac{N}{2}$. We werken het linkerlid van de vergelijking uit:

$$\begin{aligned} D_{n+1}(s, t) &= pv_{n+1}(2s) + (1-p)v_{n+1}(2s-N) - pv_{n+1}(s+t) - (1-p)v_{n+1}(s-t) \\ &= p\{p+(1-p)v_n(4s-N)\} + (1-p) \cdot 0 - p\{p+(1-p)v_n(2s+2t-N)\} - (1-p)\{pv_n(2s-2t)\} \\ &= p(1-p)\{v_n(4s-N) - v_n(2s+2t-N) - v_n(2s-2t)\}. \end{aligned}$$

□

Lemma 9.4

$$s \geq \frac{N}{2} \Rightarrow D_{n+1}(s, t) = (1-p)\{D_n(2s-N, 2t) + p(1-v_n(2s-2t))\}$$

Bewijs

We zullen eerst twee gevallen onderscheiden voor $s-t$.

- Stel eerst $s-t < \frac{N}{2}$. Eerst werken we het linkerlid uit:

$$\begin{aligned} D_{n+1}(s, t) &= pv_{n+1}(2s) + (1-p)v_{n+1}(2s-N) - pv_{n+1}(s+t) - (1-p)v_{n+1}(s-t) \\ &= p \cdot 1 + (1-p)v_{n+1}(2s-N) - p\{p+(1-p)v_n(2s+2t-N)\} - (1-p)\{pv_n(2s-2t)\} \\ &= p + (1-p)v_{n+1}(2s-N) - p^2 - p(1-p)v_n(2s+2t-N) - p(1-p)v_n(2s-2t). \end{aligned}$$

En vervolgens bekijken we het rechterlid:

$$\begin{aligned} &(1-p)\{D_n(2s-N, 2t) + p(1-v_n(2s-2t))\} \\ &= (1-p)\{pv_n(4s-2N) + (1-p)v_n(4s-3N) - pv_n(2s+2t-N) \\ &\quad - (1-p)v_n(2s-2t-N) + p(1-v_n(2s-2t))\} \\ &= p(1-p)v_n(4s-2N) + (1-p)^2v_n(4s-3N) - p(1-p)v_n(2s+2t-N) \\ &\quad - (1-p)^2 \cdot 0 + (1-p)p - p(1-p)v_n(2s-2t) \\ &= p(1-p)v_n(4s-2N) + (1-p)^2v_n(4s-3N) - p(1-p)v_n(2s+2t-N) \\ &\quad + (1-p)p - p(1-p)v_n(2s-2t). \end{aligned}$$

We moeten nu dus alleen nog het volgende te bewijzen:

$$v_{n+1}(2s-N) = pv_n(4s-2N) + (1-p)v_n(4s-3N).$$

Voor het linkerlid geldt:

$$v_{n+1}(2s-N) = \begin{cases} pv_n(4s-2N) & \text{als } \frac{N}{2} \leq s < \frac{3N}{4} \\ p + (1-p)v_n(4s-3N) & \text{als } \frac{3N}{4} \leq s \leq N \end{cases}$$

Als $\frac{N}{2} \leq s < \frac{3N}{4}$ dan geldt voor het rechterlid:

$$pv_n(4s-2N) + (1-p)v_n(4s-3N) = pv_n(4s-2N) + (1-p) \cdot 0 = pv_n(4s-2N).$$

En als $\frac{3N}{4} \leq s \leq N$ dan is het rechterlid:

$$\begin{aligned} pv_n(4s-2N) + (1-p)v_n(4s-3N) &= p \cdot 1 + (1-p)v_n(4s-3N) \\ &= p + (1-p)v_n(4s-3N). \end{aligned}$$

Dus de gelijkheid in de stelling klopt voor $s-t < \frac{N}{2}$.

- Stel nu $s-t \geq \frac{N}{2}$. Dan wordt het linkerlid:

$$\begin{aligned} D_{n+1}(s, t) &= pv_{n+1}(2s) + (1-p)v_{n+1}(2s-N) - pv_{n+1}(s+t) - (1-p)v_{n+1}(s-t) \\ &= p \cdot 1 + (1-p)v_{n+1}(2s-N) - p\{p + (1-p)v_n(2s+2t-N)\} - (1-p)\{p + (1-p)v_n(2s-2t-N)\} \\ &= (1-p)v_{n+1}(2s-N) - p(1-p)v_n(2s+2t-N) - (1-p)^2v_n(2s-2t-N). \end{aligned}$$

Als we het rechterlid uitwerken, krijgen we

$$\begin{aligned} &(1-p)\{D_n(2s-N, 2t) + p(1-v_n(2s-2t))\} \\ &= (1-p)\{pv_n(4s-2N) + (1-p)v_n(4s-3N) - pv_n(2s+2t-N) \\ &\quad - (1-p)v_n(2s-2t-N) + p(1-1)\} \\ &= p(1-p)v_n(4s-2N) + (1-p)^2v_n(4s-3N) - p(1-p)v_n(2s+2t-N) - (1-p)^2v_n(2s-2t-N). \end{aligned}$$

We zien nu dat sommige termen samenvallen en dat het voldoende is om te bewijzen:

$$v_{n+1}(2s-N) = pv_n(4s-2N) + (1-p)v_n(4s-3N).$$

Voor het linkerlid geldt:

$$v_{n+1}(2s-N) = \begin{cases} pv_n(4s-2N) & \text{als } \frac{N}{2} \leq s < \frac{3N}{4} \\ p + (1-p)v_n(4s-3N) & \text{als } \frac{3N}{4} \leq s \leq N \end{cases}$$

Stel $\frac{N}{2} \leq s < \frac{3N}{4}$, dan geldt voor het rechterlid:

$$pv_n(4s-2N) + (1-p)v_n(4s-3N) = pv_n(4s-2N) + (1-p) \cdot 0 = pv_n(4s-2N).$$

En als $\frac{3N}{4} \leq s \leq N$, dan krijgen we voor het rechterlid:

$$pv_n(4s-2N) + (1-p)v_n(4s-3N) = p \cdot 1 + (1-p)v_n(4s-3N) = p + (1-p)v_n(4s-3N).$$

We hebben nu dus de gelijkheid ook voor het geval $s-t \geq \frac{N}{2}$ bewezen.

□

Stelling 9.3

$D_n(s, t) \geq 0 \Rightarrow D_{n+1}(s, t) \geq 0$ voor alle $s, t \in \mathbb{N}_0$ met $0 \leq t \leq s \leq N$

Bewijs

We onderscheiden drie gevallen voor s en t :

- Stel $s \leq \frac{N}{2}, s + t \leq \frac{N}{2}$. Dan geldt vanwege lemma 9.2:

$$D_{n+1}(s, t) = pD_n(2s, 2t)$$

Uit $D_n(2s, 2t) \geq 0$ volgt nu meteen $D_{n+1}(s, t) \geq 0$.

- Stel $s \leq \frac{N}{2}, s + t \geq \frac{N}{2}$. lemma 9.3 geeft dan

$$D_{n+1}(s, t) = p(1 - p)\{v_n(4s - N) - v_n(2s + 2t - N) - v_n(2s - 2t)\}.$$

Uit lemma 9.1 volgt (neem $a = 2s + 2t - N \geq 0$ en $b = 2s - 2t$, dan is $a + b = 4s - N \leq N$):

$$v_n(4s - N) \geq v_n(2s + 2t - N) + v_n(2s - 2t),$$

dus dan geldt

$$v_n(4s - N) - v_n(2s + 2t - N) - v_n(2s - 2t) \geq 0.$$

Hieruit volgt onmiddellijk $D_{n+1}(s, t) \geq 0$.

- Stel $s \geq \frac{N}{2}$. Volgens lemma 9.4 hebben we nu:

$$D_{n+1}(s, t) = (1 - p)\{D_n(2s - N, 2t) + p(1 - v_n(2s - 2t))\}.$$

Omdat $D_n(2s - N, 2t) \geq 0$ en $1 - v_n(2s - 2t) \geq 0$ (de laatste ongelijkheid volgt uit $0 \leq v_n(2s - 2t) \leq 1$), krijgen we wederom $D_{n+1}(s, t) \geq 0$.

□

We kunnen nu dus concluderen dat $D_n(s, t) \geq 0$ geldt voor alle $n \in \mathbb{N}$ en $0 \leq t \leq s \leq N$ (volgt uit stellingen 9.1 en 9.3). Dus als de speler maar een eindig aantal, zeg $n + 1$, spellen mag spelen, dan is iedere keer maximaal inzetten een optimaal strategie. Aangezien dit geldt voor alle $n \in \mathbb{N}_0$, dan is maximaal inzetten ook een optimale strategie als er geen beperking is in het aantal spellen dat de speler mag spelen.

10 Simulatie

In hoofdstuk 5 hebben we bewezen dat als de winkans p in het casinomodel gelijk is aan $\frac{1}{2}$, dan is iedere strategie optimaal. De bijbehorende waarde-vector v is dan

$$v_i = \frac{i}{N}, 1 \leq i \leq N.$$

Verder hebben we in hetzelfde hoofdstuk gezien dat we ook een gesloten uitdrukking hebben voor de waarde-vector als $p > \frac{1}{2}$. In dit geval is iedere keer 1 euro inzetten de optimale strategie en is de waarde-vector v gelijk aan

$$v_i = \frac{1 - r^i}{1 - r^N} \text{ met } r = \frac{1 - p}{p}, 1 \leq i \leq N.$$

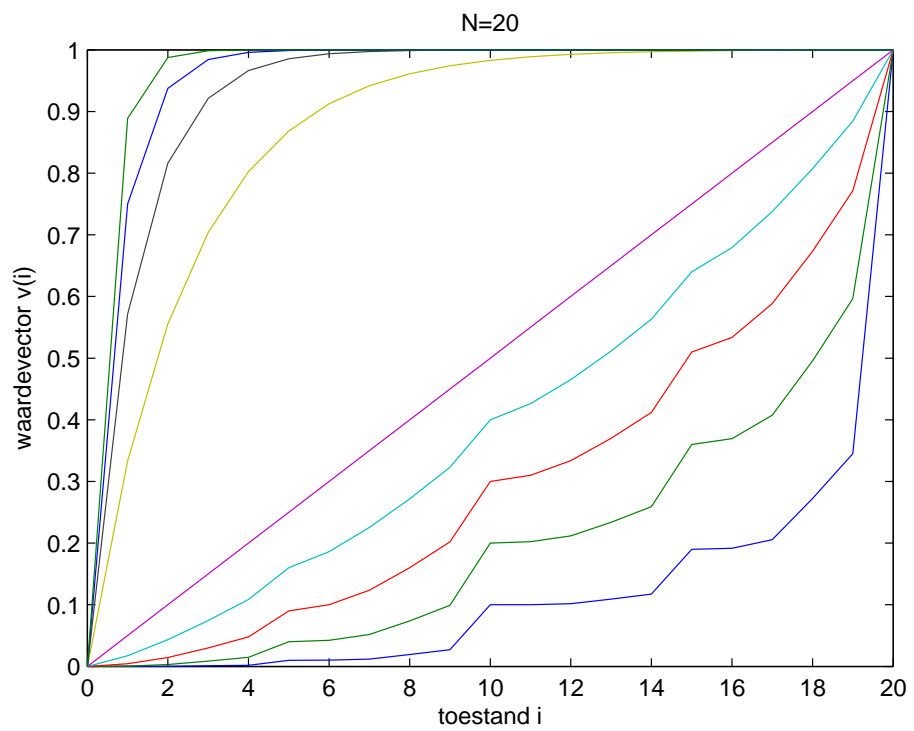
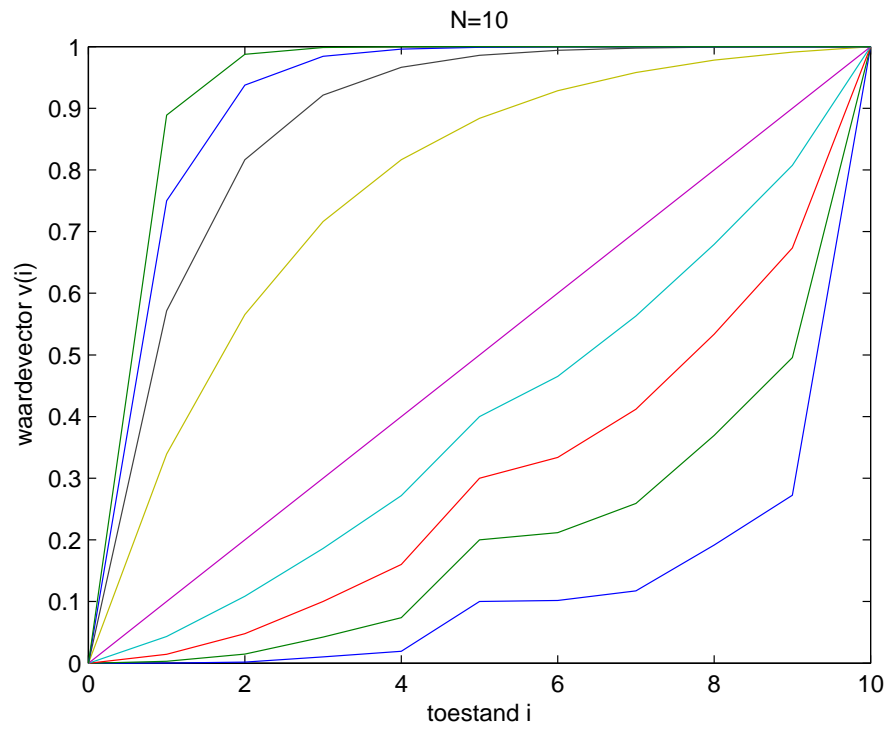
Ook hebben we voor het geval $p < \frac{1}{2}$ de optimale strategie kunnen bepalen. Nu is zoveel mogelijk inzetten de optimale strategie en dit is bewezen in hoofdstuk 9. Echter in het bewijs is de exacte waarde van de waarde-vector nooit ter sprake gekomen.

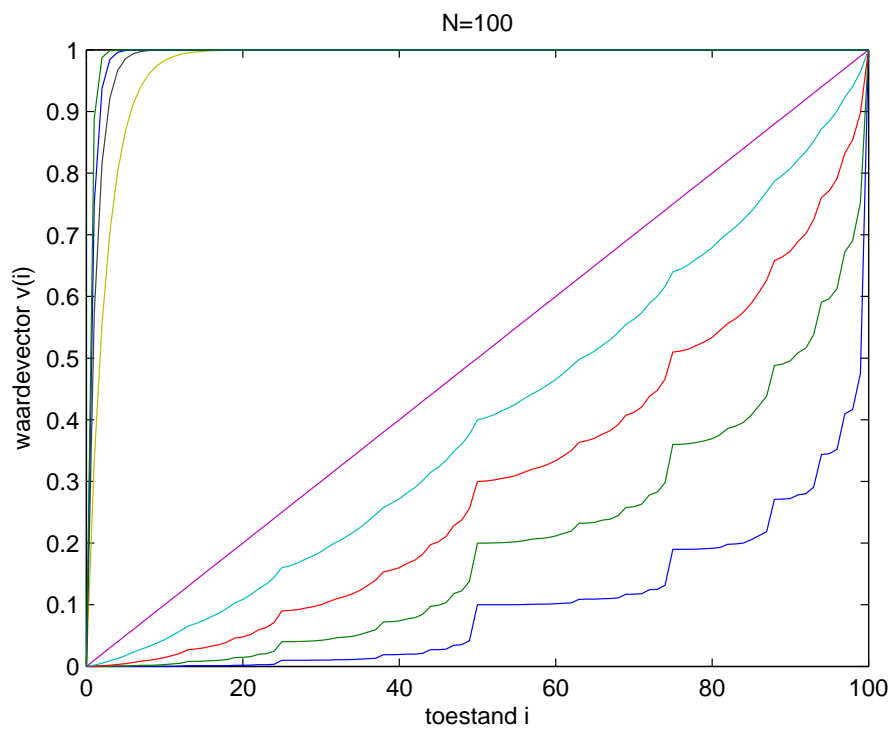
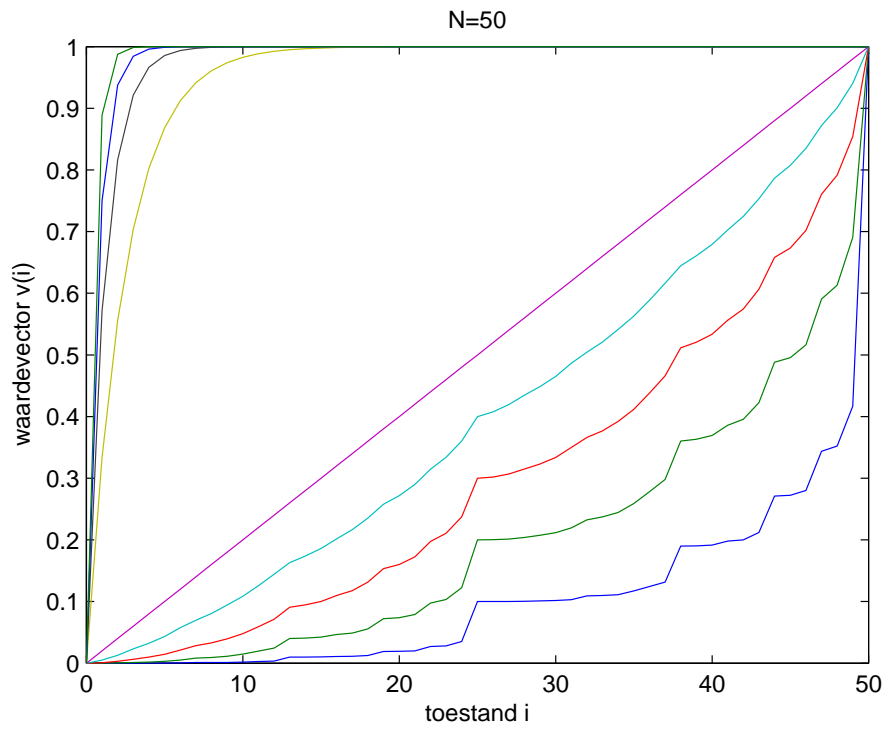
Ik denk zelf dat in het algemeen geen gesloten uitdrukking bestaan voor de waarde-vector voor het geval $p < \frac{1}{2}$. Maar gelukkig hebben we in hoofdstuk 8 (zie *Algoritme voor successieve approximatie*) gezien dat we de waarde-vector v willekeurig dichtbij kunnen benaderen. (Natuurlijk kunnen we voor een aantal toestanden vrij eenvoudig de exacte waarde bepalen, bijv. $v_0 = 0$, $v_N = 1$ en $v_{\frac{N}{2}} = p$, mits $\frac{N}{2}$ een geldig toestand is natuurlijk.)

Om wat meer inzicht te krijgen hoe groot de waarde-vector werkelijk is, heb ik een programma in C++ geschreven om de waarde-vector te benaderen voor een gegeven N , p en eps . De broncode van dit programma staat in de bijlage. In dit programma heb ik het volgende algoritme gebruikt:

1. Kies $x \in \mathbb{R}^{N+1}$ met $x_i = \frac{i}{N}$ voor $0 \leq i \leq N$.
2. (a) $y_0 := 0$ en $y_N := 1$;
(b) Bereken $y_i = \max_{a \in A(i)} \{px_{i+a} + (1-p)x_{i-a}\}$, $0 < i < N$.
3. Als $\|y - x\|_\infty \leq eps$, dan is y onze benadering van de waarde-vector v (STOP);
Anders: $x := y$ en ga naar stap 2.

Daarna heb ik voor een aantal waarden voor N (namelijk 10, 20, 50 en 100) de waarde-vector laten benaderen met het programma voor $p \in \{\frac{1}{10}, \frac{2}{10}, \frac{3}{10}, \frac{4}{10}\}$. Hierbij heb ik iedere keer 10^{-5} als eps genomen. Voor iedere waarde voor N heb ik deze uitkomsten via Matlab in een grafiek verwerkt. In deze grafieken heb ik ook de volgende waarden voor p opgenomen: $p \in \{\frac{5}{10}, \frac{6}{10}, \frac{7}{10}, \frac{8}{10}, \frac{9}{10}\}$ (voor deze p -waarden hebben we natuurlijk de exacte uitdrukking van de waarde-vector, dus was het niet nodig om deze te benaderen met het programma). Dit geeft de volgende plaatjes:



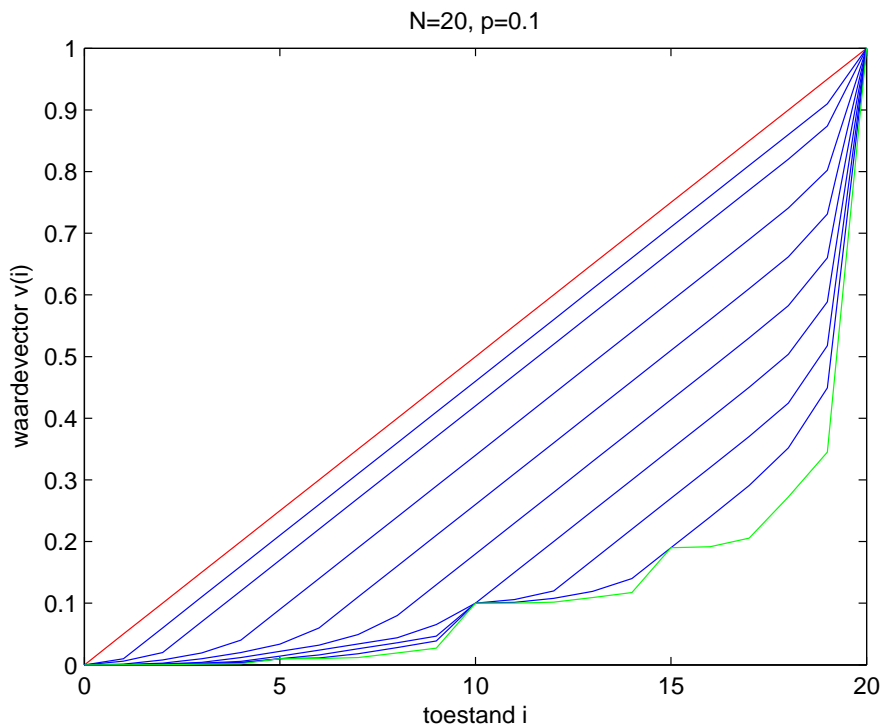


Uiteraard hoort een hogere lijn bij een hogere p -waarde in iedere afbeelding. Het valt op dat voor $p < \frac{1}{2}$ de lijn niet zo'n mooie vorm heeft, op sommige plekken

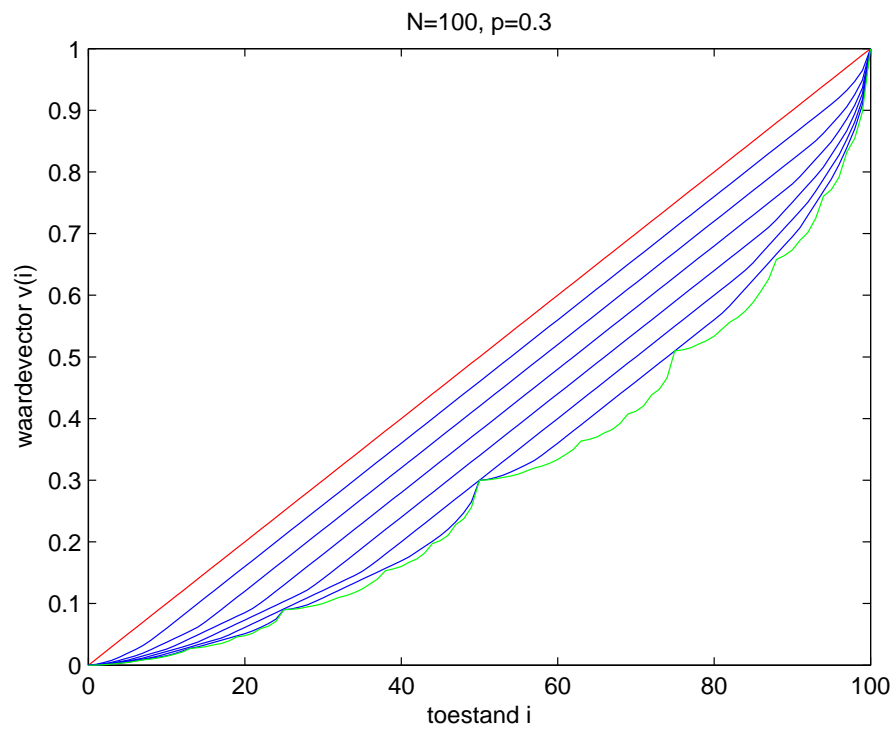
stijgt de lijn namelijk plotseling relatief veel tussen een toestand i en $i+1$. Terwijl bij $p = \frac{1}{2}$ een rechte lijn hoort en bij $p > \frac{1}{2}$ is de lijn een mooie concave functie. Je zou dan verwachten dat bij $p < \frac{1}{2}$ een convexe functie hoort. (Aan de andere kant als de lijn wel een mooie vorm zou hebben, dan zou de waarde-vector waarschijnlijk wel een gesloten uitdrukking hebben.)

Om een indruk te geven hoe snel de vector convergeert naar de exacte waarden van de waarde-vector, heb ik tenslotte voor een tweetal voorbeelden in een grafiek laten weergegeven wat de waarde is van de vector na een bepaald aantal iteraties.

In het eerste voorbeeld heb ik $N = 20$ en $p = \frac{1}{10}$ genomen. Als $eps = 10^{-5}$ geldt, dan zijn er 24 iteraties nodig. In het volgende plaatje is de startvector in het rood en de (benaderde) waarde-vector in het groen weergegeven. De lijnen in het blauw zijn de vectors verkregen na $\{1,2,4,\dots,12,14\}$ iteraties:



In het tweede voorbeeld is $N = 100$ en $p = \frac{3}{10}$. Bij $eps = 10^{-5}$ zijn in dit geval 81 iteraties nodig. Wederom zijn in de volgende grafiek de startvector en de (benaderde) waarde-vector in het rood en in het groen weergegeven. Verder zijn de lijnen (blauw) weergegeven na $\{10,20,\dots,60\}$ iteraties:



11 Conclusie

Het doel van deze scriptie was om de optimale strategie te bepalen voor de casinospeler, zodat zijn kans om het streefbedrag N te bereiken wordt gemaximaliseerd. We hebben in de hoofdstukken 2 en 3 gezien dat het casinoprobleem eenvoudig om te zetten is in een Markov beslissingsketen.

Daarna hebben we bewezen dat dit model ook transiënt is (hoofdstuk 4). Dit hadden we nodig om met een verdisconteerd model te kunnen werken. Met de theorie van de verdisconteerde opbrengsten hebben we vervolgens de optimale strategie kunnen afleiden voor de gevallen $p = \frac{1}{2}$ en $\frac{1}{2} < p < 1$ (zie hoofdstuk 5). Hierbij hebben we eerst moeten aantonen dat de bijbehorende optimaliteitsvergelijking een unieke oplossing heeft. We hebben gezien dat als $p = \frac{1}{2}$ geldt, dan is iedere strategie optimaal. En als $\frac{1}{2} < p < 1$ geldt, dan moet de speler iedere keer 1 euro inzetten.

Voor het geval $0 < p < \frac{1}{2}$ hebben we veel meer inspanning moeten verrichten om de optimale strategie te bepalen. We hebben hierbij successieve approximatie (hoofdstuk 8) gebruikt. Dit is een iteratieproces waarbij de waarde-vector willekeurig precies kan worden benaderd. We beginnen met een willekeurige vector als oplossing en in iedere stap van het iteratieproces wordt de functie U (gedefinieerd in hoofdstuk 7) toegepast op de verkregen vector. Maar dit gaat alleen goed als de functie U een uniek vaste punt heeft. Het was hierbij voldoende om aan te tonen dat U een monotone contractie afbeelding (hoofdstuk 6) is en dit hebben we bewezen in hoofdstuk 7. Vervolgens hebben we in hoofdstuk 9 kunnen bewijzen wat de optimale strategie is als $0 < p < \frac{1}{2}$. De speler moet namelijk iedere keer $\min\{i, N - i\}$ euro inzetten als zijn huidige toestand i is.

Voor het geval $\frac{1}{2} \leq p < 1$ hebben we gezien dat we de waarden van de bijbehorende waarde-vectoren exact kunnen bepalen. Dit is niet zo in het geval $0 < p < \frac{1}{2}$ (maar we hebben bewezen ook niet bewezen dat er geen exacte uitdrukking bestaat voor de waarde-vector). Daarom was het wel aardig om de waarde-vector in hoofdstuk 10 te benaderen d.m.v. simulaties voor een aantal gevallen. Het simuleren ging via successieve approximatie. Hierdoor hebben we een beter beeld gekregen hoe de waarde-vector eruit ziet als $\frac{1}{2} \leq p < 1$.

Zoals te zien is, is het bijbehorende onderzoek bij deze bachelorscriptie goed geslaagd. We hebben namelijk voor ieder waarde voor de winkans p kunnen bepalen wat theoretisch de beste strategie is. Hopelijk komt deze informatie de lezer goed van pas als hij of zij de volgende keer naar het casino gaat. Maar de lezer dient mijn aanbevelingen wel op eigen risico te gebruiken, want ik wil niet aansprakelijk worden gesteld voor de gevolgen!

Bijlage

Dit is de broncode van het C++ programma dat gebruikt is in hoofdstuk 10:

```
#include <iostream>
using namespace std;

int main ( )
{
    const int C=1000; // maximale grootte array
    int N; // toestandsruimte is {0,1,...,N}
    int i; // toestand
    int a; // actie
    int iteratie; // zoveelste iteratie
    double p; // winkans
    double eps; // max. grootte tussen x en y
    double verschilxy; // verschil tussen x en y
    double x[C]; // vector x
    double y[C]; // vector y
    int maxinzet; // wat maximaal ingezet kan worden
    bool stop=0; // moet algoritme gestopt worden?

    // gebruiker moet eerst een aantal gegevens invoeren:
    cout << "Wat is N?" << endl;
    cin >> N;
    cout << "Wat is p?" << endl;
    cin >> p;
    cout << "Geef een eps:" << endl ;
    cin >> eps;
    cout << endl;

    x[0]=0; y[0]=0; x[N]=1; y[N]=1; // deze waarden weten we toch al

    for (i=1;i<=N-1;i++) // we kunnen beginnen met willekeurige x uit  $R^N$ 
        x[i]=double (i)/N; // in dit geval nemen we  $x(i)=i/N$ 

    iteratie=0;
    while(stop==0) // iteratieproces
    {
        for (i=1;i<=N-1;i++)
        {
            if(i<=N-i) // maximale inzet wordt bepaald
                maxinzet=i;
        }
    }
}
```

```

        else
            maxinzet=N-i;

        y[i]=0;
        for(a=1;a<=maxinzet;a++) // y wordt berekent
        {   if(p*x[i+a]+(1-p)*x[i-a]>y[i])
            y[i]=p*x[i+a]+(1-p)*x[i-a];
        }
    }

    verschilxy=0;
    for (i=1;i<=N-1;i++) // norm van x-y wordt bepaald
    {   if (y[i]-x[i]>verschilxy)
        verschilxy=y[i]-x[i];
        if (x[i]-y[i]>verschilxy)
            verschilxy=x[i]-y[i];
    }

    if (verschilxy<eps) // is verschil tussen x en y klein genoeg?
        stop=1;
    else
        for (i=1;i<=N-1;i++)
            x[i]=y[i];

    iteratie++;
}

for (i=0;i<=N;i++) // resultaten worden op het scherm getoond
    cout << "v(" << i << ")=" << y[i] << endl;

cout << endl << "Er zijn " << iteratie << " iteraties uitgevoerd." << endl;

int h; cin >> h; // voorkomt dat programma direct afsluit
return 0;
}

```