

MATERIAŁY I STUDIA

Zeszyt nr 201

Programowanie dynamiczne i modele rekursywne w ekonomii.

Zagadnienia analityczne i metody numeryczne
z przykładowymi implementacjami w języku Matlab/Octave

Grzegorz Klima

Warszawa, grudzień 2005 r.

Projekt graficzny:

Oliwka s.c.

Skład i druk:

Drukarnia NBP

Wydął:

Narodowy Bank Polski
Departament Komunikacji Społecznej
00-919 Warszawa, ul. Świętokrzyska 11/21
tel. (22) 653 23 35, fax (22) 653 13 21

© Copyright Narodowy Bank Polski, 2004

Materiały i Studia rozprowadzane są bezpłatnie.

Dostępne są również na stronie internetowej NBP: <http://www.nbp.pl>

Od wydawcy

Metody programowania dynamicznego (czy też szerzej – dynamicznej optymalizacji) i modelowania dynamiki gospodarczej poprzez zależności rekursywne zajmują miejsce szczególne we współczesnej analizie ekonomicznej. Świadczy o tym mnogość anglojęzycznej literatury na ten temat i dobór artykułów w prestiżowych czasopismach ekonomicznych. Na polskim rynku wydawniczym wciąż jednak nie ma pozycji adresowanej do ekonomistów, traktującej o zagadnieniach optymalizacji dynamicznej w czasie dyskretnym i problemach stochastycznych. Mamy nadzieję, że niniejsza pozycja choć częściowo wypełni tę lukę.

Książka ta jest próbą zwięzłej prezentacji metod programowania dynamicznego. Główny nacisk jest położony na wyrobienie podstawowego warsztatu i intuicji pozwalających samodzielnie rozwiązywać, symulować i kalibrować (lub estymować) modele rekursywne. W założeniu, po lekturze niniejszej książki można bez większych trudności sięgnąć do współczesnej literatury ekonomicznej, w szczególności tej dotyczącej dynamicznych, stochastycznych modeli równowagi ogólnej (DSGE), a także podjąć samodzielne próby konstrukcji modeli tego typu. Ze względu na wybrane podejście, formalne wywody ograniczone są do minimum, zaś przykłady ekonomiczne służą głównie ilustracji opisywanych metod.

Książka jest adresowana przede wszystkim do ekonomistów pragnących rozszerzyć zestaw narzędzi analitycznych i numerycznych, którymi operują. Niniejszym pozycja powinna być przystępna dla wszystkich osób dysponujących odpowiednim, podstawowym zapleczem matematycznym i ekonomicznym.

Modele DSGE, do których niniejsza książka stanowi wprowadzenie teoretyczne i narzędziowe, nabierają dużego znaczenia nie tylko w samej teorii ekonomii, lecz także w praktyce makromodelowania. Wyestymowane lub skalibrowane modele równowagi ogólnej, mające solidne podstawy metodologiczne i teoretyczne, stanowią coraz atrakcyjniejszą alternatywę dla luźno związanych z teorią lub wręcz z założenia ateoretycznych modeli ekonometrycznych wykorzystywanych przez wiele instytucji. Osiągnięcia w zakresie modelowania procesów pieniężnych w równowadze ogólnej sprawiają, że banki centralne coraz częściej podejmują się konstrukcji i estymacji modeli DSGE. Departament Analiz Makroekonomicznych i Strukturalnych, realizując jedno z podstawowych zadań NBP jakim jest działalność edukacyjna, umożliwia poprzez wydanie tego opracowania w serii wydawniczej „Materiały i Studia” upowszechnianie wiedzy na temat programowania dynamicznego i modeli rekursywnych w ekonomii. Tym samym włącza się w ten obiecujący i coraz bardziej popularny nurt badawczy.

Czytelnik otrzymuje do ręki pierwszą z czterech części przygotowywanej książki, która traktuje o modelach deterministycznych. Stanowi ona wprowadzenie i przygotowanie do bardziej zaawansowanych zagadnień omawianych w kolejnych częściach, które także ukażą się w ramach serii „Materiały i Studia”.



 Spis treści

I Zagadnienia deterministyczne	9
1 Najprostszy problem programowania dynamicznego. Rekursja	11
1.1 Denicje	11
1.2 Problem najkrótszej drogi	13
1.3 Rekursja i równanie Bellmana	14
1.4 Dyskontowanie	18
1.5 Rekursja na komputerze	19
1.5.1 Algorytm rozwiązywania problemów z dyskretną przestrzenią stanów	19
1.5.2 Implementacja algorytmu w języku Matlab/Octave	21
1.5.3 Wycena zasobu	22
1.5.4 Problem najkrótszej drogi	24
Zadania	24
2 Ciągła przestrzeń stanów. Równania Eulera. Nieskończony horyzont	27
2.1 Twierdzenie o obwiedni i równania Eulera	27
2.1.1 Twierdzenie o obwiedni	27
2.1.2 Problem optymalnej konsumpcji majątku w czasie	29
2.1.3 Interpretacja równania Eulera i jego bezpośrednie wyprowadzenie	31
2.1.4 Metoda mnożników Lagrange'a	33
2.2 Nieskończony horyzont. Model Ramseya	35
2.2.1 Problem centralnego planisty w modelu Ramseya	35
2.2.2 Warunek transwersalności w zagadnieniach z nieskończonym horyzontem	37
2.3 Metoda kolejnych przybliżeń	37
2.4 Algorytm kolejnych przybliżeń w zagadnieniach z dyskretną przestrzenią stanów	39
2.4.1 Algorytm	39
2.4.2 Implementacja w języku Matlab/Octave	40
Zadania	42
3 Warunki konieczne i dostateczne optymalności sterowania	43
3.1 Denicje i założenia	43
3.2 Od optymalnej polityki do równania Bellmana	45
3.3 Dostateczność równania Bellmana	48
3.4 Dostateczność równania Eulera w połączeniu z warunkiem transwersalności	51
3.5 Uwagi	53

Zadania	54
4 Interakcje optymalizujących podmiotów. Równowaga ogólna	55
4.1 Czysta wymiana w dwóch okresach	55
4.1.1 <i>Problem jednego konsumenta</i>	55
4.1.2 <i>Dwóch konsumentów. Alokacje. Efektywność</i>	57
4.1.3 <i>Równowaga konkurencyjna</i>	59
4.2 Wymiana w horyzoncie nieskończonym. Równowaga Arrowa-Debreu i rekursywna	62
4.2.1 <i>Wymiana w okresie $t = 0$</i>	62
4.2.2 <i>Równowaga rekursywna</i>	64
4.3 Twierdzenia ekonomii dobrobytu	67
4.3.1 <i>Założenia i denicje</i>	68
4.3.2 <i>Pierwsze twierdzenie ekonomii dobrobytu</i>	71
4.3.3 <i>Drugie twierdzenie ekonomii dobrobytu</i>	72
4.3.4 <i>Uwagi</i>	73
4.4 Centralny planista i koncepcja reprezentatywnego podmiotu	75
Zadania	76
5 Konsumpcja i inwestycje w równowadze ogólnej	77
5.1 Podstawowe założenia	77
5.2 Podstawowa wersja modelu	79
5.2.1 <i>Wymiana w okresie $t = 0$</i>	80
5.2.2 <i>Równowaga rekursywna</i>	82
5.2.3 <i>Uwaga na temat konwencji czasowej</i>	84
5.3 Stan ustalony	84
5.4 Egzogeniczny wzrost ludności i postęp technologiczny	86
5.5 Wprowadzenie do wyceny aktywów	90
Zadania	93
6 Metoda kolokacji	95
6.1 Metody projekcyjne i metoda kolokacji	95
6.2 Przykład zastosowania metody kolokacji	96
6.3 Równanie Eulera jako równanie funkcyjne	98
6.4 Numeryczne rozwiązanie modelu Ramseya	99
Zadania	103
Dodatki	105
A Podstawy obsługi programów Matlab i Octave	107
A.1 Operacje arytmetyczne. Zmienne	107

A.2 Wektory i macierze	108
A.2.1 Deklaracje i przypisania wektorów i macierzy	109
A.2.2 Wektory i macierze specjalne	110
A.2.3 Działania na wektorach i macierzach	111
A.3 Programowanie w języku Matlab/Octave	114
A.3.1 Skrypty i funkcje	114
A.3.2 Konstrukcje warunkowe i pętle	117
A.3.3 Parę uwag o dobrym stylu programowania	121
A.4 Kilka dodatkowych możliwości programów Matlab i Octave	122
A.4.1 Wczytywanie i zapisywanie danych z/do plików	122
A.4.2 Łączenie wektorów	123
A.4.3 Operacje na zmiennych tekstowych	123
A.4.4 Tworzenie wykresów	124
A.4.5 Sprawdzanie czasu wykonania operacji	124
A.4.6 Funkcja feval	125
A.4.7 Opcjonalne argumenty funkcji. Zmienna nargin	126
B Elementy metod numerycznych	129
B.1 Podstawy. Układy równań liniowych	129
B.1.1 Reprezentacja zmiennoprzecinkowa liczb w komputerze	129
B.1.2 Liczba operacji	130
B.1.3 Układy równań liniowych	130
B.2 Aproksymacja funkcji. Wielomiany Czebyszewa	135
B.2.1 Podstawowy problem aproksymacji wielomianami	135
B.2.2 Dwie denicje wielomianów Czebyszewa	136
B.2.3 Węzły Czebyszewa	138
B.3 Rozwiązywanie równań nieliniowych	140
B.3.1 Równania nieliniowe w jednym wymiarze	140
B.3.2 Układy równań nieliniowych	143
B.3.3 Kilka uwag na temat rozwiązywania równań nieliniowych	148
C Podstawy analizy funkcjonalnej	149
C.1 Przestrzenie Banacha	149
C.1.1 Przestrzenie liniowe	149
C.1.2 Przestrzenie unormowane	150
C.1.3 Metryka i zbieżność	151
C.1.4 Ciągi Cauchy'ego i zupełność	153
C.1.5 Odwzorowania zblizające	156
C.1.6 Przestrzenie l_p	158
C.2 Odwzorowania liniowe	162
C.2.1 Odwzorowania liniowe ciągłe	162

C.2.2 Funkcjonały liniowe i przestrzeń dualna	164
C.3 Wypukłość	166
Bibliografia	171
Indeks	175

Część I
Zagadnienia deterministyczne

Rozdział 1

Najprostszy problem programowania dynamicznego. Rekursja

An optimal policy has the property that, whatever the initial state and decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision.

Richard E. Bellman, [6]

1.1 Definicje

Celem naszym jest zgłębianie metod optymalizacji, a więc szukanie najlepszych wariantów w sensie wartości pewnej funkcji celu. „Wariantami” mogą być elementy bardzo różnych zbiorów – ciągi elementów zbioru skończonego, punkty w \mathbb{R}^N , ciągi nieskończone, funkcje rzeczywiste czy wreszcie procesy stochastyczne. Określenie przestrzeni możliwych wariantów jak i funkcji (funkcjonału) celu jest istotą konstrukcji modelu ekonomicznego, który chcemy rozwiązać. My zaczniemy od nazwania tego, czego – mając już sformułowany model – szukamy.

Definicja 1.1 *Niech f będzie pewnym odwzorowaniem zbioru X w \mathbb{R} . Przez $\max_{x \in X} \{f(x)\}$ rozumiemy takie M (jeśli istnieje), że:*

$$\forall x \in X f(x) \leq M \wedge \exists x_0 \in X f(x_0) = M.$$

Definicja 1.2 *Niech X i f j.w. Przez $\min_{x \in X} \{f(x)\}$ rozumiemy takie m (jeśli istnieje), że:*

$$\forall x \in X f(x) \geq m \wedge \exists x_0 \in X f(x_0) = m.$$

Definicja 1.3 *Niech X i f j.w. Jeśli $\max_{x \in X} \{f(x)\}$ istnieje to oznaczamy:*

$$\arg \max_{x \in X} \{f(x)\} \equiv \{y \in X : f(y) = \max_{x \in X} \{f(x)\}\} \quad (1.1)$$

1.1 Definicje

$\arg \min_{x \in X} \{f(x)\}$ definiuje się analogicznie, jako podzbiór na którym przyjmowana jest wartość najmniejsza.

W niniejszej książce zajmujemy się tylko maksymalizacją. Wszystkie problemy minimalizacyjne można zamieniać na maksymalizacyjne, co wynika z poniższego lematu (wystarczy przyjąć $a = -1$).

Lemat 1.1 Niech X i f j.w. Dla $a > 0$ zachodzą równości (zakładamy, że wszystkie wyrażenia mają sens):

$$\max_{x \in X} \{af(x)\} = a \max_{x \in X} \{f(x)\}, \quad (1.2)$$

oraz

$$\arg \max_{x \in X} \{af(x)\} = \arg \max_{x \in X} \{f(x)\}, \quad (1.3)$$

Gdy $a < 0$ jest:

$$\min_{x \in X} \{af(x)\} = a \max_{x \in X} \{f(x)\}, \quad (1.4)$$

oraz

$$\arg \min_{x \in X} \{af(x)\} = \arg \max_{x \in X} \{f(x)\}, \quad (1.5)$$

Ćwiczenie 1.1 Udowodnij powyższe równości.

Ćwiczenie 1.2 Pokaż, że jeśli f i g są funkcjami z X w \mathbb{R} to:

$$\max_{x \in X} \{f(x) + g(x)\} \leq \max_{x \in X} \{f(x)\} + \max_{x \in X} \{g(x)\} \quad (1.6)$$

i

$$\min_{x \in X} \{f(x) + g(x)\} \geq \min_{x \in X} \{f(x)\} + \min_{x \in X} \{g(x)\}. \quad (1.7)$$

Ćwiczenie 1.3 Pokaż, że jeśli f i g są funkcjami z X w \mathbb{R} i $\forall x f(x) \geq g(x)$ to:

$$\max_{x \in X} \{f(x)\} \geq \max_{x \in X} \{g(x)\} \quad (1.8)$$

i

$$\min_{x \in X} \{g(x)\} \leq \min_{x \in X} \{f(x)\}. \quad (1.9)$$

Ćwiczenie 1.4 Pokaż, że jeśli f jest funkcją z X w \mathbb{R} i $A \subseteq B \subseteq X$ to:

$$\max_{x \in B} \{f(x)\} \geq \max_{x \in A} \{f(x)\} \quad (1.10)$$

i

$$\min_{x \in B} \{f(x)\} \leq \min_{x \in A} \{f(x)\} \quad (1.11)$$

Udowodnimy teraz pewien ważny lemat.

1 Najprostszy problem programowania dynamicznego. Rekursja

Lemat 1.2 Niech $f: X \rightarrow \mathbb{R}$ i $g: X \times Y \rightarrow \mathbb{R}$. Dodatkowo niech dla każdego $x \in X$ istnieje $\max_y \{g(x, y)\}$, a także istnieje $\max_{(x,y)} \{f(x) + g(x, y)\}$. Wtedy:

$$\max_{(x,y)} \{f(x) + g(x, y)\} = \max_x \{f(x) + \max_y \{g(x, y)\}\}.$$

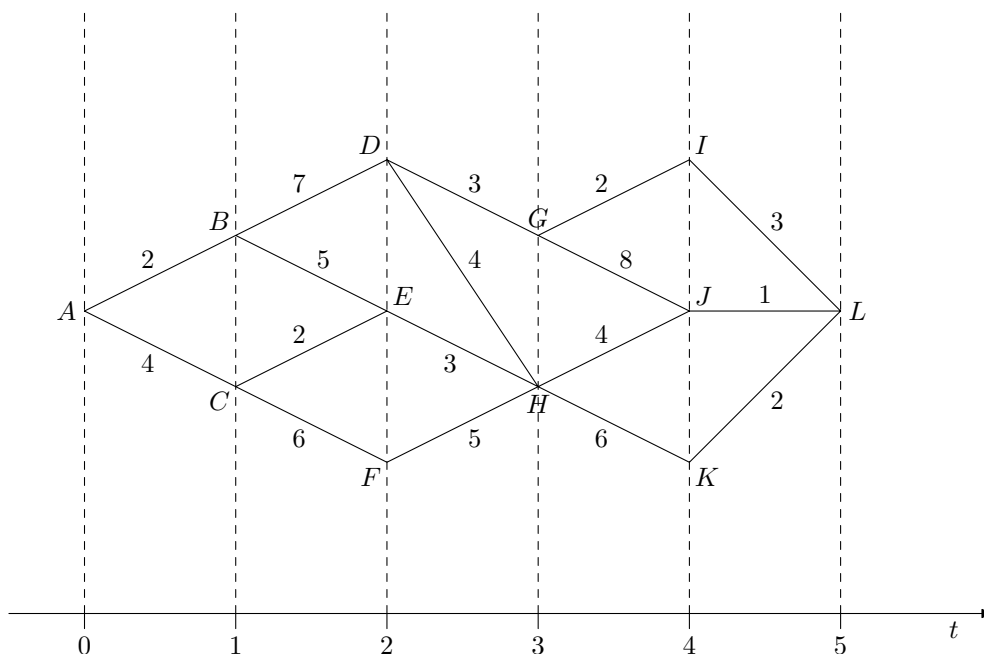
Dowód. Niech L i R oznaczają lewą i prawą stronę równości odpowiednio. Pokażemy, że $R \leq L$ i $R \geq L$.

Z definicji prawej strony istnieje takie x' , że $f(x') + \max_y \{g(x', y)\} = R$ i istnieje takie y' , że $g(x', y') = \max_y \{g(x', y)\}$, czyli $\exists_{(x',y')} f(x') + g(x', y') = R$. Z definicji lewej strony wynika, że $R \leq L$.

Zauważmy, że $\max_{(x,y)} \{f(x) + \max_y \{g(x, y)\}\} = \max_x \{f(x) + \max_y \{g(x, y)\}\} = R$. Ponieważ $\forall_{(x,y)} f(x) + \max_y \{g(x, y)\} \geq f(x) + g(x, y)$ mamy $R \geq \max_{(x,y)} \{f(x) + g(x, y)\} = L$. ■

1.2 Problem najkrótszej drogi

Rozpatrzmy problem znalezienia najkrótszej drogi z punktu A do L na „mapie” z rysunku 1.1.¹ Każda z dróg wiedzie przez 5 odcinków (6 skrzyżowań), przy czym każdemu odcinkowi przyporządkowana jest pewna długość, ew. czas przejazdu. Na problem nieco sztucznie nakładamy strukturę czasową zakładając, że w każdym kolejnym okresie kierowca dojeżdża do następnego skrzyżowania.



Rysunek 1.1: Problem najkrótszej drogi

Korzystając z lematu 1.1 problem ten zamienimy na zagadnienie maksymalizacji przyporządkowując każdemu odcinkowi wypłatę równą długości trasy wziętej

¹ Przykład ten prezentujemy za Chiangiem [8].

1.3 Rekursja i równanie Bellmana

z minusem.² Tak więc X jest zbiorem możliwych dróg, a odwzorowanie f przyporządkowuje każdej z nich liczbę będącą jej długością wziętą z minusem. Mamy:

$$X = \{ABDGIL, ABDGJL, ABDHJL, ABDHKL, ABEHJL, \\ ABEHKL, ACEHJL, ACEHKL, ACFHJL, ACFHKL\}$$

i

$$f(ABDGIL) = -17, f(ABDGJL) = -21, f(ABDHJL) = -18, f(ABDHKL) = -21, \\ f(ABEHJL) = -15, f(ABEHKL) = -18, f(ACEHJL) = -14, f(ACEHKL) = -17, \\ f(ACFHJL) = -20, f(ACFHKL) = -23.$$

Porównanie wypłat pozwala nam stwierdzić, że:

$$\max_{x \in X} f(x) = -14$$

i

$$\arg \max_{x \in X} \{f(x)\} = \{ACEHJL\}.$$

Rozwiązanie naszego problemu jest jednoznaczne: najkrótsza trasa wiedzie przez punkty A, C, E, H, J i L .

Zauważmy, że powyższa metoda rozwiązywania problemu jest bardzo żmudna i dla większych zagadnień nieaplikowalna. Dodatkowo nieoczywista jest jej implementacja komputerowa a to ze względu problem „wypisania” wszystkich możliwych ścieżek.³

Ideą przewodnią naszego rozwiązania było porównywanie „całych” dopuszczalnych trajektorii, a więc poszukiwania rozwiązania na pewnym zbiorze ciągów. Jeśli zbiór X ma odpowiednią strukturę (jest np. przestrzenią Banacha) a funkcja f jest różniczkowalna to porównywanie elementów „dużych” przestrzeni (np. ciągów czy funkcji) daje się zastąpić rozwiązaniem równań nakładanych przez warunek zerowania się pochodnej (klasyczne podejście analityczne). Podejście to „zabija” sekwencyjną strukturę problemu kosztem powiększenia (uogólnienia) przestrzeni poszukiwań. Jak zobaczyliśmy powyżej czasami taka metoda nie jest zbyt efektywna, nawet w przypadku skrajnie (na pierwszy rzut oka) prostego zagadnienia.

Poniżej prezentujemy podejście rekursywne, tj. wprost odwołujące się do sekwencyjnej struktury problemu. Jak później się okaże w miejscach stosowalności klasycznych metod pokrywa się ono z nimi. Jednak nasz prosty przykład jest świetną okazją do wprowadzenia podstawowych pojęć i wyrobienia intuicji.

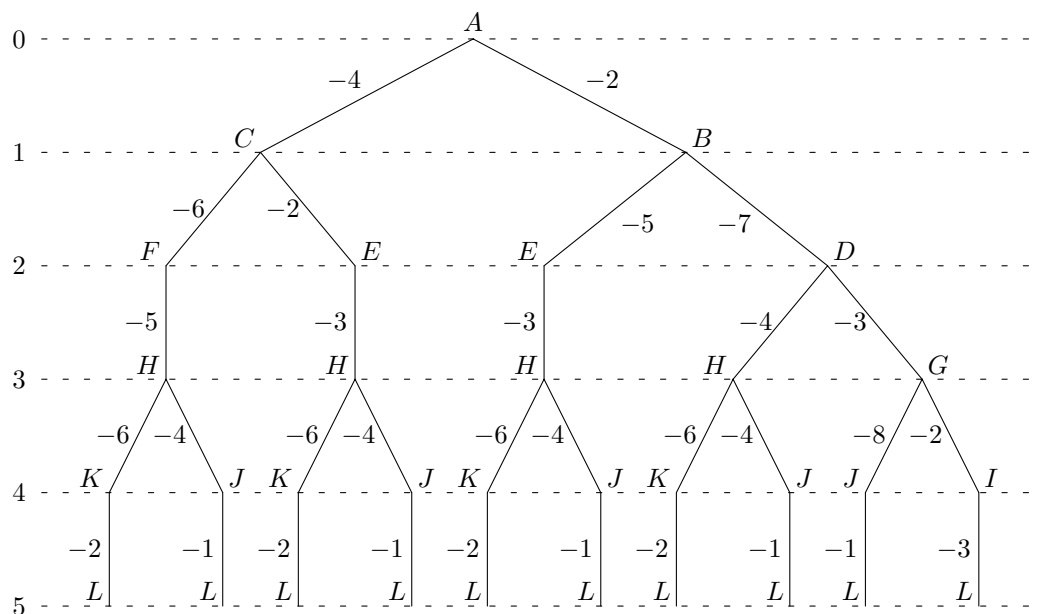
1.3 Rekursja i równanie Bellmana

Celem uwypuklenia sekwencyjnej struktury problemu najkrótszej drogi narysujmy drzewo decyzyjne kierowcy (rys. 1.2). Wierzchołki odpowiadają skrzyżowaniom, ramiona drzewa możliwym trasom.

² Można traktować ten problem jak problem maksymalizacji użyteczności, w którym droga jest „złem”.

³ Tak naprawdę już w tym momencie pojawia się potrzeba rekursji.

1 Najprostsz problem programowania dynamicznego. Rekursja



Rysunek 1.2: Drzewo decyzyjne w problemie najkrótszej drogi

Wprowadzimy teraz dwa ważne określenia. Przez stan będziemy rozumieli egzogeniczne (w momencie podejmowania decyzji) warunki ograniczające, zaś przez sterowanie (politykę, ang. *policy*) rozumiemy decyzję podejmowaną w danym momencie. W problemie o sekwencyjnej strukturze dzisiejsza polityka w połączeniu z dzisiejszym stanem określają stan jutrzejszy a więc jutrzejsze ograniczenia (np. aktywa netto, skrzyżowanie, na którym się znajdziemy itp.). Stany będziemy od tej pory oznaczać przez s , zaś sterowania przez c (ang. *control*). Jeśli np. jesteśmy w stanie $s = A$, dostępne sterowania to $\{AB, AC\}$. Zbiór sterowań dostępnych w stanie s będziemy oznaczać przez $C(s)$. Odwzorowanie określające stan s_{t+1} , w jakim znajdziemy się wybierając w okresie t , w stanie s_t sterowanie $c_t \in C(s_t)$ oznaczmy przez g i będziemy określać mianem funkcji przejścia. Równanie:

$$s_{t+1} = g(s_t, c_t), \quad (1.12)$$

będziemy nazywać równaniem dynamiki lub krócej: dynamiką zagadnienia.

Fundamentalną cechą wszystkich problemów dających się rozwiązywać przy pomocy równań Bellmana jest rozdzielność funkcji celu w czasie. Innymi słowami, funkcja celu musi być sumą wkładów z kolejnych okresów przy czym wkład w każdym okresie zależy tylko od stanu i wybranego sterowania:

$$\sum_{t=0}^T u_t(s_t, c_t) + u_{T+1}(s_{T+1}).$$

Ostatni człon określa użyteczność stanu końcowego, tj. stanu w okresie $T+1$, w której nie podejmujemy już żadnych decyzji.

W naszym przykładzie funkcja celu jest sumą wziętych z minusem długości odcinków odpowiadających sterowaniom.

1.3 Rekursja i równanie Bellmana

Założmy, że znajdujemy się w stanie s_t i decydujemy się na wybór pewnego ciągu sterowań $c_t \in C(s_t), c_{t+1} \in C(s_{t+1}), \dots, c_T \in C(s_T)$, przy czym przez kolejne stany przechodzimy zgodnie z dynamiką zagadnienia (1.12). Wyrażenie:

$$U(s_t, c_t, c_{t+1}, \dots, c_T) = \sum_{i=t}^T u_i(s_i, c_i) + u_{T+1}(s_{T+1})$$

w naturalny sposób określa wypłatę związaną z przyjęciem strategii c_t, c_{t+1}, \dots, c_T . Wprowadźmy teraz podstawowe pojęcie:

Definicja 1.4 Funkcją wartości $V_t(s_t)$ nazywamy odwzorowanie przypisujące każdemu stanowi maksymalną osiągalną wypłatę:

$$\max_{c_t, c_{t+1}, \dots, c_T} \left\{ \sum_{i=t}^T u_i(s_i, c_i) + u_{T+1}(s_{T+1}) \right\}$$

$$\text{pw. } \forall_{i \in \{t, t+1, \dots, T\}} c_i \in C(s_i) \wedge s_{i+1} = g(s_i, c_i).$$

W szczególności funkcja wartości od stanu początkowego ($V_0(s_0)$) to maksymalna osiągalna wartość funkcji celu, zaś $V_{T+1}(s_{T+1}) = u_{T+1}(s_{T+1})$.

Wykorzystując pojęcie funkcji wartości możemy skonstruować rekurencyjny algorytm rozwiązania naszego problemu. Jeśli jesteśmy w punkcie L problem decyzyjny jest trywialny – jesteśmy na końcu drogi i przyjmujemy $V(L) = 0$. Założmy więc, że dojechaliśmy do punktu I . W tej sytuacji mamy tylko jedną strategię do wyboru: sterowanie IL , z którym wiąże się wypłata -3 , czyli $V(I) = -3$. Analogicznie $V(J) = -1$ – optymalna (jedyna możliwa) polityka JL i $V(K) = -2$ – optymalna polityka KL . Jeśli teraz cofniemy się o jeden krok, możemy określić, ile warte jest bycie w stanach G i H . Jeśli ze stanu G pójdziemy do I uzyskamy -2 plus w najlepszym przypadku $V(I) = -3$ czyli -5 . Jeśli z G pójdziemy do J – -9 . Tak więc $V(G) = \max\{-5, -9\} = -5$. Analogicznie $V(H) = -5$. Wiemy już, że jeśli dojdziemy do G najlepsza trasa to GIL związana z wypłatą -5 , zaś gdy dojdziemy do H – optymalna trasa to HJL z wypłatą -5 . Jeśli jesteśmy w stanie D najlepszą strategią jest pójście do G . Pobyt w stanie D gwarantuje wypłatę -8 . W ten sposób możemy się cofać aż do znalezienia optymalnej trasy z A , czyli do rozwiązania. Rysunek 1.3 przedstawia pełne rozwiązanie problemu. W drzewie pogrubiono optymalne ścieżki wychodzące z każdego wierzchołka.

Zauważmy, że na każdym etapie wyznaczania funkcji wartości posługiwaliśmy się następującym równaniem:

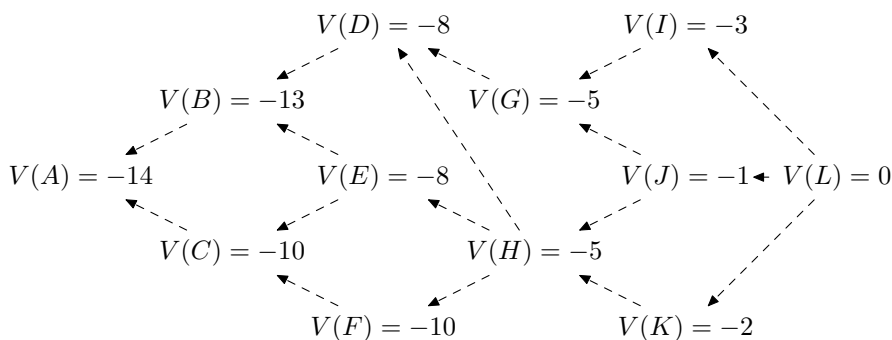
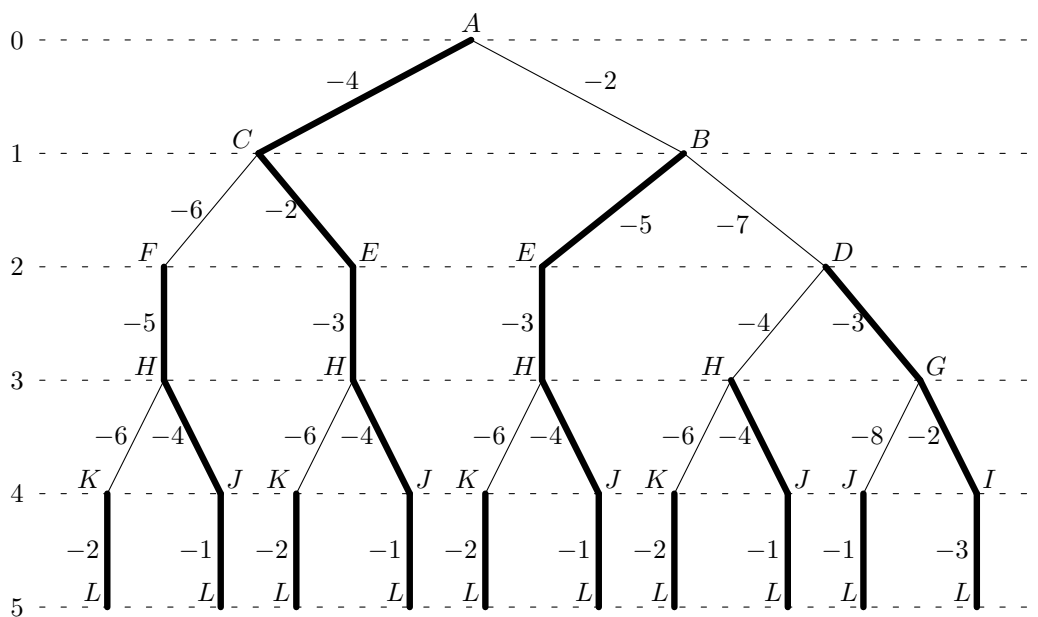
$$V_t(s_t) = \max_{c_t \in C(s_t)} \{u_t(s_t, c_t) + V_{t+1}(s_{t+1}(s_t, c_t))\}. \quad (1.13)$$

Równanie (1.13) nosi nazwę równania Bellmana.

Optymalne sterowanie (ze stanu s_t) $c^*(s_t)$ opisane jest tzw. równaniem polityki (ang. *policy equation*):

$$c_t^*(s_t) = \arg \max_{c_t \in C(s_t)} \{u_t(s_t, c_t) + V_{t+1}(s_{t+1}(s_t, c_t))\}. \quad (1.14)$$

1 Najprostszy problem programowania dynamicznego. Rekursja



Rysunek 1.3: Rozwiązanie problemu najkrótszej drogi

Równanie Bellmana można też wyprowadzić wprost. Załóżmy, że jesteśmy w stanie s_t . Z definicji funkcji wartości mamy:

$$V_t(s_t) = \max_{c_t, c_{t+1}, \dots, c_T} \left\{ \sum_{i=t}^T u_i(s_i, c_i) + u_{T+1}(s_{T+1}) \right\}$$

Korzystając z lematu 1.2 uzyskujemy:

$$V_t(s_t) = \max_{c_t, c_{t+1}, \dots, c_T} \left\{ \sum_{i=t}^T u_i(s_i, c_i) + u_{T+1}(s_{T+1}) \right\} = \max_{c_t} \left\{ u_t(s_t, c_t) + \max_{c_{t+1}, \dots} \left\{ \sum_{i=t+1}^T u_i(s_i, c_i) + u_{T+1}(s_{T+1}) \right\} \right\}.$$

Wyrażenie $\max_{c_{t+1}, \dots} \left\{ \sum_{i=t+1}^T u_i(s_i, c_i) + u_{T+1}(s_{T+1}) \right\}$ jest funkcją wartości pewnego



1.4 Dyskontowanie

jutrzejszego stanu s_{t+1} będącego wynikiem dzisiejszej decyzji⁴ c_t . Zauważmy, że niezależnie od tej decyzji, pozostałe decyzje (a więc c_{t+1}, \dots, c_T) muszą być optymalne ze względu na wartość $\sum_{i=t+1}^T u_i(s_i, c_i) + u_{T+1}(s_{T+1})$ pod warunkiem wyjściowego stanu s_{t+1} . Jest to cytowana na początku rozdziału zasada optymalności Bellmana. Podsumowując nasze rozumowanie uzyskujemy równanie (1.13).

1.4 Dyskontowanie

Wszystkie modele ekonomiczne bazujące na wielookresowej optymalizacji uwzględniają dyskontowanie przyszłych zysków ew. przyszłej użyteczności. Zadanie podmiotu gospodarującego ma więc postać:

$$\max \left\{ \sum_{t=0}^T \beta^t u(s_t, c_t) + \beta^{T+1} u_{T+1}(s_{T+1}) \right\}, \quad (1.15)$$

przy założeniu pewnej dynamiki $s_{t+1} = g(s_t, c_t)$. Zakładamy, że $\beta \in (0, 1)$ ($\beta^{-1} - 1$ ma interpretację stopy dyskontowej).

Dla tego szczególnego przypadku zmienimy definicję funkcji wartości. Poniższa definicja będzie już nas obowiązywała do końca książki.

Definicja 1.5 Funkcją wartości $V_t(s_t)$ w problemie z dyskontowaniem (1.15) nazywamy odwzorowanie przypisujące każdemu stanowi maksymalną osiągalną wypłatę dyskontowaną na dany okres:

$$\max \left\{ \sum_{i=t}^T \beta^{i-t} u(s_i, c_i) + \beta^{T+1-t} u_{T+1}(s_{T+1}) \right\}$$

pw. $\forall_{i \in \{t, t+1, \dots, T\}} c_i \in C(s_i) \wedge s_{i+1} = g(s_i, c_i)$.

Wyprowadzimy teraz równanie Bellmana w zagadnieniu z dyskontowaniem. Załóżmy jak poprzednio, że jesteśmy w stanie s_t . Z (nowej) definicji funkcji wartości mamy:

$$V_t(s_t) = \max_{c_t, c_{t+1}, \dots, c_T} \left\{ \sum_{i=t}^T \beta^{i-t} u_i(s_i, c_i) + \beta^{T+1-t} u_{T+1}(s_{T+1}) \right\}$$

Korzystając najpierw z lematu 1.2 a potem z lematu 1.1 uzyskujemy:

$$\begin{aligned} V_t(s_t) &= \max_{c_t, c_{t+1}, \dots, c_T} \left\{ \sum_{i=t}^T \beta^{i-t} u_i(s_i, c_i) + \beta^{T+1-t} u_{T+1}(s_{T+1}) \right\} = \\ &= \max_{c_t} \left\{ u_t(s_t, c_t) + \max_{c_{t+1}, \dots} \left\{ \sum_{i=t+1}^T \beta^{i-t} u_i(s_i, c_i) + \beta^{T+1-t} u_{T+1}(s_{T+1}) \right\} \right\} = \\ &= \max_{c_t} \left\{ u_t(s_t, c_t) + \beta \max_{c_{t+1}, \dots} \left\{ \sum_{i=t+1}^T \beta^{i-t-1} u_i(s_i, c_i) + \beta^{T-t} u_{T+1}(s_{T+1}) \right\} \right\} = \end{aligned}$$

⁴ Właśnie fakt, iż dzisiejsze decyzje wpływają na jutrzejszy stan zmuszał nas do korzystania z lematu 1.2.

1 Najprostsz problem programowania dynamicznego. Rekursja

$$\max_{c_t} \{u(s_t, c_t) + \beta V_{t+1}(s_{t+1}(s_t, c_t))\}.$$

Ostatecznie, stwierdzamy, że równanie Bellmana w rozpatrywanym problemie ma postać:

$$V_t(s_t) = \max_{c_t} \{u(s_t, c_t) + \beta V_{t+1}(s_{t+1}(s_t, c_t))\}. \quad (1.16)$$

Optymalna polityka opisana jest poniższym równaniem:

$$c_t^*(s_t) = \arg \max_{c \in C(s_t)} \{u(s_t, c_t) + \beta V_{t+1}(s_{t+1}(s_t, c_t))\}. \quad (1.17)$$

Rozpatrzmy teraz następujący przykład wyceny zasobu naturalnego.⁵ Pod ziemią znajduje się \bar{s} ton rudy. W każdym roku można wybrać całkowitą liczbę ton rudy,⁶ przy czym jeśli aktualnie w złożu jest s ton rudy, to wydobyć c ton wiąże się z kosztem $c^2/(1+s)$. Zakładamy, że cena, po której sprzedawana jest ruda, nie zmienia się w czasie i wynosi p . Stopa procentowa, według której dyskontowane są przyszłe zyski, jest równa r . Horyzont czasowy, w którym przewiduje się wydobyć, to T lat.

Ćwiczenie 1.5 Napisz równanie ewolucji zasobu ($s_{t+1}(s_t, c_t)$) i funkcję celu w tym problemie.

W powyższym problemie zmienną stanu jest wielkość zasobu, zaś sterowaniem – wydobyć w danym roku. Współczynnik dyskontujący jest równy $(1+r)^{-1}$. Równanie Bellmana ma postać:

$$V_t(s_t) = \max_{c_t \in \{0, 1, 2, \dots, s_t\}} \left\{ pc_t - \frac{c_t^2}{1+s_t} + \frac{1}{1+r} V_{t+1}(s_{t+1}) \right\}.$$

Wartość naszego złoża to oczywiście $V_0(\bar{s})$.

1.5 Rekursja na komputerze

1.5.1 Algorytm rozwiązywania problemów z dyskretną przestrzenią stanów

Celem naszym będzie teraz skonstruowanie algorytmu rozwiązywania problemów z dyskretną przestrzenią stanów i skończonym horyzontem. Ponumerujmy stany od 1 do n ($s \in \{1, 2, \dots, n\}$) i sterowania od 1 do m ($c \in \{1, 2, \dots, m\}$). Nasze rozwiązanie problemu najkrótszej drogi rozpoczynało się „na końcu” od określenia wypłaty związanej ze stanem, w jakim znajdowaliśmy się w okresie T . Punktem wyjścia naszego algorytmu powinien być więc wektor $n \times 1$ końcowych wypłat związanych z pobytami w okresie $T+1$ w każdym ze stanów $s_{1,2,\dots,n}$. Oznaczmy ten wektor przez v . Tak więc $v_i = u_{T+1}(s_i)$. Wejściem naszego algorytmu powinna być także macierz R

⁵ Przykład ten pochodzi od Mirandy i Facklera [16].

⁶ To założenie jest po prostu dyskretyzacją przestrzeni stanu, tzn. zastąpieniem problemu z ciągłym stanem i ciągłym sterowaniem przez problem ze skończoną liczbą stanów i sterowań. Oczywiście oczekujemy, że rozwiązanie zdyskretyzowanej wersji problemu przybliży rozwiązanie problemu pierwotnego i to tym lepiej, im gęstszy jest podział przestrzeni stanu.

1.5 Rekursja na komputerze

(ang. *reward*) $n \times m$ wypłat przypisującą każdemu stanowi i każdemu sterowaniu liczbę $R_{i,j} = u(c_j, s_i)$, a także macierz TR (ang. *transition*) $n \times m$ przejścia opisywająca w jakim stanie się znajdziemy, jeśli teraz ze stanu i wybierzemy sterowanie j .

Niech T oznacza nasz horyzont czasowy (przyjmujemy, że czas zaczyna się od jedynek – zmiana ta ma oczywiście charakter czysto techniczny). Chcemy znaleźć macierz $n \times T$ funkcji wartości V i macierz $n \times T$ sterowań C , które opisują odpowiednio funkcję wartości i optymalne sterowanie ze stanu i w okresie j . Macierz V rozszerzymy do rozmiarów $n \times (T + 1)$ tak, by w ostatnią kolumnę wpisać wektor v (to też są funkcje wartości – w okresie $T + 1$).

Założmy, że znamy wszystkie funkcje wartości w okresie t (wypełniliśmy t -tą kolumnę macierzy V). Wtedy funkcję wartości stanu j w okresie $t - 1$ możemy wyznaczyć korzystając z równania Bellmana (1.16):

$$V_{t-1,i} = \max_{j \in \{1, \dots, m\}} \{R_{i,j} + \beta V_{t,TR_{i,j}}\}$$

Optymalne sterowanie uzyskujemy z równania polityki (1.17):

$$C_{t-1,i} = \arg \max_{j \in \{1, \dots, m\}} \{R_{i,j} + \beta V_{t,TR_{i,j}}\}$$

Do obliczeń przydatny będzie wektor p , w którym będziemy zapisywali (dla ustalonego czasu i stanu) wypłaty $R_{i,j} + \beta V_{t,TR_{i,j}}$ z każdej z polityk.

Oto pełny algorytm:

1. Wczytaj β , T , macierze R , TR i wektor v .
2. Inicjalizuj macierze C i V . Wpisz w ostatnią kolumnę V wektor v .
3. Ustal $t = T$.
4. Ustal stan $s = 1$.
5. Ustal sterowanie $c = 1$.
6. Przypisz $p_c = R_{s,c} + \beta V_{t+1,TR_{s,c}}$
7. Jeśli $c < m$, zwiększ c o jeden i idź do 6.
8. Przypisz $V_{s,t} = \max_i \{p_i\}$ i $C_{s,t} = \arg \max_i \{p_i\}$.
9. Jeśli $s < n$, zwiększ s o jeden i idź do 5.
10. Jeśli $t > 1$, zmniejsz t o jeden i idź do 4.

Konstruując nasz algorytm milcząco założyliśmy, że z każdego stanu dostępne jest każde z m sterowań. Co zrobić jeśli tak nie jest? Wystarczy zastosowanie tzw. metody kar, czyli wpisanie $-\infty$ w odpowiednie miejsca macierzy R . Dzięki temu zabiegowi niedopuszczalne sterowanie nie zostanie nigdy wybrane.

1.5.2 Implementacja algorytmu w języku Matlab/Octave

Nasz algorytm działa w trzech pętłach. W pierwszej startujemy w okresie T i kolejno dla $T, T-1, \dots, 1$ wyznaczamy funkcję wartości dla każdego ze stanów (pętla druga). W trzeciej, środkowej pętli wyznaczamy wypłaty z każdej z możliwych polityk (wektor `policy_values`) i znajdujemy najlepszą. W tym miejscu wykorzystujemy informację o stopie dyskontującej β (`beta`). Do znajdowania najlepszej polityki i wypłaty z niej wykorzystujemy funkcję `max`. Wprowadzamy następujące oznaczenia: t – `t`, T – `time`, R – `reward`, TR – `transition`, v – `terminal`, β – `discount`, C – `control`, V – `value`, p – `policy_values`. Oto kod naszego programu:

```

_____ Plik dprecur1.m _____
function [C, V]=dprecur1(time, discount, reward, transition, terminal)
%function [C, V]=dprecur1(time, discount, reward, transition, terminal)
%
%dprecur1 rozwiązuje rekursywnie problem programowania dynamicznego
%dla skończonej liczby stanów i skończonego horyzontu czasowego
%
%time - horyzont czasowy
%discount - współczynnik dyskontujący
%reward - macierz wypłat
%transition - macierz przejścia
%terminal - wektor (kolumnowy) wypłat końcowych
%
%(c) Grzegorz Klima 2004

[n, m]=size(reward);

control=zeros(n, time);
value=zeros(n, time+1);
value(:,time+1)=terminal;

policy_values=zeros(1,m);

for t=time:-1:1
    for i=1:n
        for j=1:m
            policy_values(j)=reward(i,j)+discount*value(transition(i,j),t+1);
        end
        [value(i,t), control(i,t)]=max(policy_values);
    end
end

C=control;
V=value;

```

_____ Koniec pliku dprecur1.m _____

Wykorzystując wyjście z naszego programu (macierz sterowań `C`) i macierz przejścia możemy napisać funkcję służącą do symulacji ewolucji zmiennej stanu przy optymalnym sterowaniu. Oto ona:

1.5 Rekursja na komputerze

```

Plik sim1.m
function ret=sim1(x0, transition, control)
%function ret=sim1(x0, transition, control)
%
%sim1 symuluje ewolucje zmiennej stanu
%
%x0 - stan poczatkowy
%transition - macierz przejścia
%control - macierz sterowan
%
%(c) Grzegorz Klima 2004

time=size(control,2);
x=zeros(1,time+1);
x(1)=x0;

for t=1:time
    x(t+1)=transition(x(t), control(x(t),t));
end

ret=x;
Koniec pliku sim1.m

```

1.5.3 Wycena zasobu

Korzystając z funkcji `dprecur1` i `sim1` rozwiązujemy problem wyceny zasobu i optymalnej polityki eksploatacyjnej. Załóżmy, że zasób początkowy wynosi $\bar{s} = 100$ ton, cena $p = 1$, zaś stopa procentowa 11,11%. Stany numerujemy od 1 do 101 przy czym i -ty stan oznacza $i - 1$ ton zasobu. Podobnie numerujemy sterowania (wydobycie). Wektor wypłat końcowych to oczywiście 101×1 zer:

```
octave:1> term = zeros(101,1);
```

Jak wyglądają macierze przejścia i wypłat? Jeśli jesteśmy w stanie i i wybieramy sterowanie j to mamy $i - 1$ rudy przy wydobyciu $j - 1$, a więc zostaje $i - j$ ton, czyli jesteśmy w stanie $i - j + 1$. Zakładamy, że jeśli wydobędziemy więcej niż mamy, to trafiamy do stanu 1. W tym ostatnim przypadku wypłata jest równa $-\infty$ (stosujemy metodę kar, by wyeliminować niedopuszczalne sterowania). Przy wydobyciu nieprzekraczającym zasobów wypłatę ustalamy zgodnie ze wzorem $(j - 1) - (j - 1)^2 / (1 + (i - 1)) = (j - 1) - (j - 1)^2 / i$. Poniżej prezentujemy kod, który inicjalizuje macierze przejścia i wypłat:

```

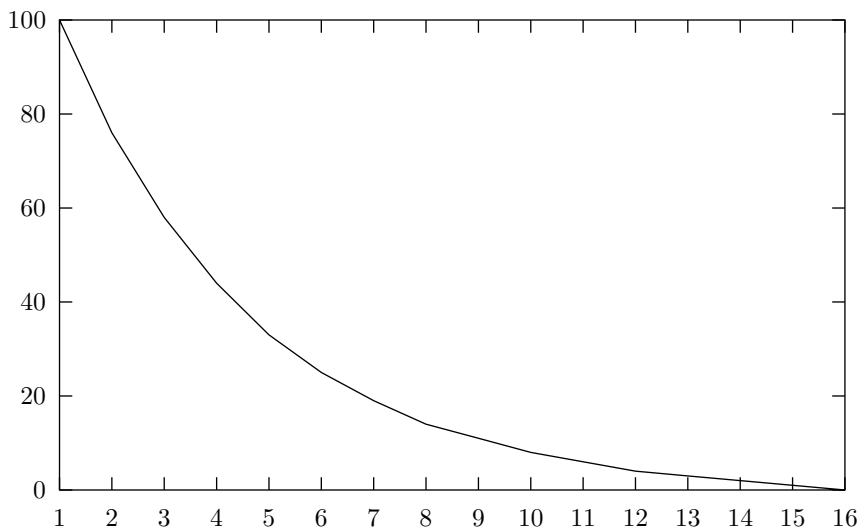
octave:2> trans = zeros(101,101);
octave:3> rew = zeros(101,101);
octave:4> for i=1:101
> for j = 1:101
> trans(i,j) = max(0,i-j)+1;
> if (i>=j) rew(i,j) = (j-1)-(j-1)^2/i; else rew(i,j) = -inf; end
> end
> end

```

1 Najprostsz y problem programowania dynamicznego. Rekursja

Wystarczy teraz uruchomić naszą funkcję `dprecu1`. Przyjmujemy horyzont czasowy 15 lat. Współczynnik dyskontowy to $(1 + r)^{-1} = 0,9$:

```
octave:5> [c,v]=dprecu1 (15, .9, rew, trans, term);
```



Rysunek 1.4: Ewolucja poziomu zasobu naturalnego przy optymalnej polityce eksploatacyjnej

Funkcja `sim1` pozwoli nam śledzić ewolucję zasobu rudy (pamiętamy o odjęciu jedyńki od numeru stanu):

```
octave:6> sim1(101, trans, c)-1
ans =

Columns 1 through 13:

   100    76    58    44    33    25    19    14    11    8    6    4    3

Columns 14 through 16:

    2    1    0
```

Ćwiczenie 1.6 Sporządź wykres wielkości wydobycia w każdym roku.

Ile warte jest złożo? Jego wartość jest funkcją wartości stanu 101 w okresie 1:

```
octave:7> v(101,1)
ans = 58.114
```

Zadania

1.5.4 Problem najkrótszej drogi

Wykorzystamy nasze dwie funkcje do rozwiązania problemu najkrótszej trasy. Skrzyżowania numerujemy według kolejności liter w alfabecie. Zauważamy, że z każdego skrzyżowania prowadzą najwyżej dwie drogi („w lewo” i „w prawo”), a więc mamy dwa możliwe sterowania (oczywiście moglibyśmy jak wcześniej rozpatrywać sterowania AB, AC, \dots , ale w ten sposób utrudnilibyśmy sobie zadanie).⁷ W przypadkach, w których jest tylko jedna możliwa droga przyjmujemy, że sterowanie nr 2 prowadzi do tego samego punktu. Podobnie z punktu L (stan 12) oba sterowania prowadzą do L . Macierz przejścia wygląda więc następująco:

```
octave:8> trans=[2 4 5 7 8 8 9 10 12 12 12 12; 3 5 6 8 5 6 10 11
9 10 11 12]'
```

W macierzy wypłat w miejscach odpowiadających niedopuszczalnym sterowaniom wpisujemy $-\text{Inf}$ ($-\infty$). Poniższa komenda inicjalizuje macierz wypłat:

```
octave:9> rew=[-2 -7 -2 -3 -3 -5 -2 -3 -4 -3 -1 -2 -Inf; -4 -5 -6
-4 -Inf -Inf -8 -6 -Inf -Inf -Inf -Inf]'
```

Na koniec drogi chcemy się znaleźć w punkcie L , więc wektor końcowych wypłat powinien wyglądać jak niżej:

```
octave:10> term=[-Inf -Inf -Inf -Inf -Inf -Inf -Inf -Inf -Inf -Inf
-Inf 0]'
```

Ćwiczenie 1.7 Jak szybciej (i łatwiej) zainicjalizować wektor `term`?

Wywołujemy funkcję `dprecur1`:

```
octave:11> [c,v]=dprecur1(5, 1, rew, trans, term);
```

Korzystając z funkcji `sim1` sprawdzamy optymalną trasę z punktu 1 (A):

```
octave:12> sim1(1, trans, c)
ans =
```

```
1 3 5 8 10 12
```

Zadania

Zadanie 1.1 Rozpatrz problem Robinsona żyjącego na bezludnej wyspie. Jedynym dobrem jest zboże. W każdym okresie Robinson ma do dyspozycji s ton zboża, z których może skonsumować c ton a resztę zasadzić. Z jednej tony zasadzonego zboża uzyskuje dwie tony plonów, jednak nie więcej niż 40 (wyspa jest mała). Robinson czerpie użyteczność z konsumpcji daną wzorem $u(c) = \ln(c)$, jego indywidualna stopa dyskontowa θ wynosi 10%. Robinson przewiduje, że będzie żył jeszcze 40 lat, w momencie wylądowania na wyspie zebrał $\bar{s} = 10$ ton zboża.

⁷ Inicjalizacja odpowiednich macierzy i wektorów w tym przykładzie jest i tak nieco żmudna.

1 Najprostsz y problem programowania dynamicznego. Rekursja

1. Zapisz problem optymalizacyjny Robinsona (funkcj e celu i r oównanie ewolucji zasobu).
2. Zapisz r oównanie Bellmana.
3. Rozwi aź problem Robinsona na komputerze. Napisz skrypty odpowiednio wype łniaj acej macierze przej scia i wyp łat. Sporz aď wykres poziomu zasob o w (zasianego zboża) i konsumpcji w czasie.

Zadanie 1.2 Wycen Ź zas o b naturalny z przyk ładu zak ładaj acej, Źe w ł aściciel zasobu jest monopolist a i w kaŹdym okresie spotyka si e z popytem $q = a - bp$.

1. Zapisz problem optymalizacyjny.
2. Zapisz r oównanie Bellmana.
3. Rozwi aź zadanie na komputerze. Napisz skrypty odpowiednio wype łniaj acej macierze przej scia i wyp łat. Przyjmij $\bar{s} = 100$, $T = 15$, $r = 11,11\%$, $a = 64$ i $b = 40$.

Rozdział 2

Ciągła przestrzeń stanów. Równania Eulera. Nieskończony horyzont

2.1 Twierdzenie o obwiedni i równania Eulera

2.1.1 Twierdzenie o obwiedni

Zacniemy od udowodnienia pewnego bardzo użytecznego twierdzenia. Rozpatrzmy pewną funkcję $f(x; \alpha)$, w której α odgrywa rolę parametru. Załóżmy, że f jest różniczkowalna w sposób ciągły względem x i α oraz dla każdego α z pewnego interesującego nas przedziału istnieje $\max_x \{f(x; \alpha)\}$. Oznaczmy¹ $\bar{x}(\alpha) \equiv \arg \max_x \{f(x; \alpha)\}$ i $F(\alpha) \equiv \max_x \{f(x; \alpha)\}$. $F(\alpha)$ jest po prostu funkcją wartości zależącą od parametru. Załóżmy dalej, że $\bar{x}(\alpha)$ i $F(\alpha)$ są różniczkowalne. Interesuje nas wpływ α na funkcję wartości. Mówi o nim następujące twierdzenie.²

Twierdzenie 2.1 (O obwiedni) *Przyjmijmy oznaczenia jak wyżej. Zachodzi następująca równość:*³

$$\frac{d}{d\alpha} F(\alpha) = f_2(\bar{x}(\alpha); \alpha). \quad (2.1)$$

Dowód.

$$\frac{d}{d\alpha} F(\alpha) = \frac{d}{d\alpha} f(\bar{x}(\alpha); \alpha) = f_1(\bar{x}(\alpha); \alpha) \frac{d}{d\alpha} \bar{x}(\alpha) + f_2(\bar{x}(\alpha); \alpha).$$

Z warunku pierwszego rzędu maksymalizacji $f(x; \alpha)$ po x wynika, że $f_1(\bar{x}(\alpha); \alpha) = 0$, co daje tezę. ■

Ćwiczenie 2.1 Przyjmij $f(x; \alpha) = -x^2 + \alpha x$. Znajdź $\bar{x}(\alpha)$ i $F(\alpha)$. Pokaż, że zachodzi (2.1).

¹ Dokonujemy tutaj pewnego nadużycia symboliki, tj. utożsamiamy (z założenia) jednoelementowy zbiór $\arg \max_x \{f(x; \alpha)\}$ z jego elementem.

² Ang. *envelope theorem* często błędnie tłumaczone jako „twierdzenie o kopercie”.

³ Przez f_1 i f_2 oznaczamy pochodne cząstkowe po pierwszym i drugim argumente odpowiednio.

2 Ciągła przestrzeń stanów. Równania Eulera. Nieskończony horyzont

Przed przystąpieniem do wykorzystania powyższego twierdzenia w zagadnieniach programowania dynamicznego pokażemy pewien prosty przykład zagadnienia wyboru optymalnego koszyka dóbr. Przy okazji przypomnimy interpretację mnożników Lagrange'a jako „cen cieni”, do której jeszcze w tym rozdziale będziemy się odwoływać.

Rozpatrzmy problem konsumenta, który dysponuje zasobem w i dokonuje wyboru między konsumpcją dwóch dóbr x_1 i x_2 . Ceny tych dóbr to p_1 i p_2 odpowiednio. Konsument maksymalizuje swą użyteczność $u(x_1, x_2) = \ln x_1 + \ln x_2$ pod warunkiem ograniczenia budżetowego $p_1x_1 + p_2x_2 \leq w$. Jest jasne, że konsument wyda swój cały majątek w (jeśli tylko $u_1 > 0$ i $u_2 > 0$), tak więc wiążącym warunkiem jest warunek $p_1x_1 + p_2x_2 = w$. Standardową metodą rozwiązywania tego typu zagadnień jest skorzystanie z funkcji Lagrange'a (λ jest pewnym nieokreślonym mnożnikiem):

$$\mathcal{L} = \ln x_1 + \ln x_2 + \lambda(w - p_1x_1 - p_2x_2).$$

Warunki pierwszego rzędu mają postać:

$$\frac{\partial \mathcal{L}}{\partial x_1} = \frac{1}{x_1} - \lambda p_1 = 0,$$

$$\frac{\partial \mathcal{L}}{\partial x_2} = \frac{1}{x_2} - \lambda p_2 = 0$$

i

$$\frac{\partial \mathcal{L}}{\partial \lambda} = w - p_1x_1 - p_2x_2 = 0.$$

Rozwiązanie powyższego układu równań daje:

$$x_1 = \frac{w}{2p_1}, \quad x_2 = \frac{w}{2p_2}, \quad \lambda = \frac{2}{w}.$$

Ćwiczenie 2.2 Dokonaj pominiętych obliczeń.

Zastanówmy się teraz nad wpływem majątku w na maksymalną osiągalną użyteczność przy zadanych cenach (oznaczymy ją $\tilde{u}(w)$). Skoro konsument zakupi $w/(2p_1)$ dobra pierwszego i $w/(2p_2)$ drugiego tu uzyska użyteczność:

$$\tilde{u} = \ln \frac{w}{2p_1} + \ln \frac{w}{2p_2}$$

Proste różniczkowanie daje:

$$\frac{d\tilde{u}}{dw} = \frac{2}{w}.$$

Tak więc zwiększenie majątku o jednostkę da (w liniowym przybliżeniu) przyrost użyteczności konsumenta o $2/w$.

Metoda mnożników Lagrange'a pozwala sprowadzić problem optymalizacji warunkowej do bezwarunkowej maksymalizacji lagranżanu. Jednocześnie wiemy, że \mathcal{L} osiąga ekstremum w punkcie $(x_1^*, x_2^*, \lambda^*)$ spełniającym warunek ograniczenia budżetowego $p_1x_1^* + p_2x_2^* = w$. Ponieważ:

$$\max_{x_1, x_2, \lambda} \{\mathcal{L}(x_1, x_2, \lambda, w)\} \equiv \mathcal{L}(x_1^*, x_2^*, \lambda^*, w) =$$

2.1 Twierdzenie o obwiedni i równania Eulera

$$\ln x_1^* + \ln x_2^* + \lambda \underbrace{(w - p_1 x_1^* - p_2 x_2^*)}_{=0} = \ln x_1^* + \ln x_2^*,$$

możemy zapisać:⁴

$$\tilde{u}(w) = \max_{x_1, x_2, \lambda} \{ \mathcal{L}(x_1, x_2, \lambda, w) \}.$$

Z twierdzenia o obwiedni uzyskujemy natychmiast:⁵

$$\frac{d\tilde{u}}{dw} = \lambda = \frac{2}{w},$$

co jest zgodne z poprzednim wynikiem.

Ostatnie równanie nadaje ważną interpretację mnożnikowi Lagrange’a⁶ — jest on pochodną funkcji wartości (maksymalnej osiągalnej przy danym poziomie zasobów użyteczności) po zasobie, a więc wycenia zasób w jednostkach użyteczności informując, że przyrost zasobu o Δw , da (w liniowym przybliżeniu) przyrost użyteczności o $\lambda \Delta w$.

2.1.2 Problem optymalnej konsumpcji majątku w czasie

Wykorzystamy teraz twierdzenie 2.1 do prostego zagadnienia programowania dynamicznego — problemu optymalnej ścieżki konsumpcji w czasie. Rozpatrzmy problem konsumenta dysponującego początkowym zasobem a_0 . W każdym z okresów od 0 do T podejmuje decyzję ile wydać na konsumpcję c a ile oszczędzić. Oszczędności procentują z okresu na okres według stopy r . Aktywa na „koniec świata”, tj. w okresie $T + 1$ muszą być nieujemne — konsument nie może zostawić po sobie długów.⁷ Konsument maksymalizuje zdyskontowany na dziś (stopa dyskontowa θ) strumień użyteczności $u(c_t)$. O funkcji u ⁸ zakładamy, że jest określona na $(0, \infty)$ i dwukrotnie różniczkowalna, przy czym $u' > 0$ oraz $u'' < 0$. Problem konsumenta można zapisać następująco⁹ (przyjmujemy $\beta \equiv (1 + \theta)^{-1}$):

$$\max \sum_{t=0}^T \beta^t u(c_t) \quad \text{p.w.} \quad a_{t+1} = (a_t - c_t)(1 + r), \quad a_{T+1} \geq 0, \quad \forall_t c_t \geq 0. \quad (2.2)$$

Fakt, że przestrzeń stanów nie jest dyskretna w żaden sposób nie zmienia naszego rozumowania prowadzącego do równania (1.13). Równanie Bellmana w omawianym zagadnieniu ma więc postać:

$$V_t(a_t) = \max_{c_t} \{ u(c_t) + \beta V_{t+1}(\underbrace{(a_t - c_t)(1 + r)}_{a_{t+1}}) \}. \quad (2.3)$$

⁴ Nie wypisujemy cen jako argumentów funkcji Lagrange’a, bo przyjmujemy je tutaj jako ustalone parametry.

⁵ Pamiętajmy, że pochodna cząstkowa po prawej stronie wzoru (2.1) jest liczona w optimum, a w naszym zadaniu w ekstremum mamy $\lambda = 2/w$.

⁶ Przy przyjętej konwencji zapisu ograniczenia w funkcji Lagrange’a. Tzn. jeśli zapisalibyśmy $\mathcal{L} = \ln x_1 + \ln x_2 + \lambda(p_1 x_1 + p_2 x_2 - w)$, to odpowiednią interpretację miało by nie λ , lecz $-\lambda$.

⁷ Wynika z tego wprost ograniczenie na sterowanie: $c_t \leq a_t$.

⁸ Jest to tzw. funkcja użyteczności chwilowej.

⁹ Zwróćmy uwagę na przyjętą konwencję czasową: a_t oznacza aktywa na początek „dnia” t , c_t — konsumpcję tego „dnia”; aktywa na koniec „dnia” procentują przez „noc”. Jeśli przyjęlibyśmy konwencję według której a_t oznaczałoby aktywa na koniec „dnia”, wtedy równanie ewolucji zasobu miałyby postać $a_{t+1} = a_t(1 + r) - c_{t+1}$ z zasobem początkowym danym przez a_{-1} .



2 Ciągła przestrzeń stanów. Równania Eulera. Nieskończony horyzont

Po „końcu świata” tj. w okresie $T + 1$ nie można już czerpać żadnej użyteczności z aktywów, tak więc $V_{T+1}(a) = 0$.¹⁰ Optymalne sterowanie w okresie T jest równe a_T . Z tego wprost uzyskujemy $V_T(a_T) = u(a_T)$. Celem naszym będzie teraz znalezienie związku wiążącego sterowania w okresach $t + 1$ i t . Jeśli założymy, że maksymalizowane wyrażenie po prawej stronie równania Bellmana (2.3) jest różniczkowalne,¹¹ możemy zapisać następujący warunek pierwszego rzędu:

$$u'(c_t) - \beta(1 + r)V'_{t+1}(a_{t+1}) = 0. \quad (2.4)$$

Zróżniczkujemy teraz równanie (2.3) po a_t . Na mocy twierdzenia o obwiedni mamy:

$$V'_t(a_t) = \beta(1 + r)V'_{t+1}(a_{t+1}). \quad (2.5)$$

Po podstawieniach $V'_{t+1}(a_{t+1}) = u'(c_t)/(\beta(1 + r))$ i $V'_t(a_t) = u'(c_{t-1})/(\beta(1 + r))$ wynikających z równania (2.4) do równania (2.5) uzyskujemy:

$$u'(c_{t-1}) = \beta(1 + r)u'(c_t).$$

Ostatecznie przesunięcie się o okres do przodu daje:

$$u'(c_t) = \beta(1 + r)u'(c_{t+1}). \quad (2.6)$$

Równanie (2.6) wiążące optymalne sterowania dziś i jutro nazywane jest zwyczajowo równaniem Eulera¹² (patrz dalej). Równania takie są tylko warunkami pierwszego rzędu, tj. nie określają jednoznacznie optymalnego sterowania. Jeśli wiemy, że optymalna konsumpcja w okresie 0 wynosi c_0^* to z równania (2.6) uzyskamy optymalną konsumpcję w okresach $1, \dots, T$. Cała trudność w korzystaniu z równań Eulera sprowadza się do określenia c_0^* . Po rozwikłaniu (2.6) można uzyskać funkcję $c_{t+1}(c_t)$. Wstawiając ją do równania dynamiki, uzyskamy pewną zależność $a_{T+1}(c_0)$. W połączeniu z warunkiem $a_{T+1} = 0$ (ze wcześniejszych rozważań wynika, że pozostawianie po sobie dodatniego majątku nie jest optymalne) określimy c_0^* .¹³ Równanie wyznaczające c_0^* nie musi mieć (i najczęściej nie ma) rozwiązania analitycznego, jednak w przypadku niektórych postaci funkcji u lub przy założeniu $\theta = r$ uzyskuje się je bez większych trudności.

Przyjmijmy $\theta = r$. Równanie (2.6) redukuje się w takim przypadku do:

$$u'(c_t) = u'(c_{t+1}).$$

¹⁰ Zostawianie po sobie majątku nie jest optymalne, gdyż dowolne sterowanie $(c_t)_{t=0}^T$ prowadzące do $a_{T+1} > 0$ można poprawić zwiększając konsumpcję w ostatnim okresie do a_T . Ponieważ krańcowa użyteczność z konsumpcji z każdego okresu jest dodatnia (bo $u' > 0$), taka zmiana rzeczywiście daje poprawę.

¹¹ Dowód, że tak rzeczywiście jest, pomijamy.

¹² Euler rozwiązując problemy dynamicznej optymalizacji w czasie ciągłym metodami, które dały początek rachunkowi wariacyjnemu, uzyskał jako warunek pierwszego rzędu równanie różniczkowe opisujące ewolucję optymalnego sterowania.

¹³ Rozumowanie powyższe stanowi podstawę dość prymitywnej metody numerycznej zwanej w literaturze anglojęzycznej *shooting* (strzelanie). Startujemy od pewnego $c_0^{(0)}$, wyliczamy $a_1^{(0)}$, rozwiązujemy (być może numerycznie) (2.6) ze względu na $c_1^{(0)}$ itd., aż dojdziemy do pewnego $a_{T+1}^{(0)}$. Dokonujemy poprawek w c_0 , aż uzyskamy po i próbach $c_0^{(i)}$ dające $a_{T+1}^{(i)}$ satysfakcjonująco bliskie zeru.

2.1 Twierdzenie o obwiedni i równania Eulera

Ponieważ u' jest funkcją monotoniczną (malejącą, bo $u'' < 0$) wnioskujemy, iż $c_t = c_{t+1}$, czyli $\forall_{t \in \{0, \dots, T\}} c_t = c$. Wcześniej stwierdziliśmy, że $c_T = a_T$. Stąd i z równania ewolucji zasobu mamy:

$$a_{T-1} = c + \frac{a_T}{1+r} = c + \frac{c}{1+r}.$$

Cofając się dalej uzyskujemy:

$$a_t = c \sum_{i=t}^T \frac{1}{(1+r)^{T-i}}.$$

Ćwiczenie 2.3 Przeprowadź indukcyjny dowód powyższej równości.

Ostatecznie c jest wyznaczone przez równanie:

$$c = a_0 \left(\sum_{t=0}^T \frac{1}{(1+r)^{T-t}} \right)^{-1} = a_0 \frac{r(1+r)^T}{(1+r)^{T+1} - 1}.$$

Oczywiście mając zadane równanie ruchu a_t i rozwikłaną (z równania Eulera) postać zależności $c_{t+1}(c_t)$ (w rozpatrywanym zagadnieniu $c_{t+1} = c_t$) moglibyśmy rozwiązywać problem idąc „do przodu” (tak, jak to wcześniej było opisane) i odwołując się na koniec do warunku $a_{T+1} = 0$.

Ćwiczenie 2.4 Znajdź optymalny poziom konsumpcji c zgodnie z powyższą sugestią. Korzystając z równania opisującego ewolucję aktywów wyznacz a_t jako funkcję a_0 i c . Rozwiąż ze względu na c równanie $a_{T+1} = 0$.

Jako ćwiczenia pozostawiamy problem alokacji dochodu w czasie w przypadku $\theta \neq r$, gdy funkcja użyteczności chwilowej ma postać pozwalającą na pełne analityczne rozwiązanie problemu (2.2).

Ćwiczenie 2.5 Rozwiąż zagadnienie (2.2) przyjmując $u(c) = \ln c$.

Ćwiczenie 2.6 J.w. Przyjmij $u(c) = \sqrt{c}$.

2.1.3 Interpretacja równania Eulera i jego bezpośrednie wyprowadzenie

Zatrzymajmy się jeszcze przez chwilę na interpretacji równania Eulera (2.6). Po przeformułowaniu ma ono postać:

$$\frac{\beta^{t+1} u'(c_{t+1})}{\beta^t u'(c_t)} = \frac{1}{(1+r)}.$$

Wyraz po prawej stronie $(1+r)^{-1}$ jest ceną względną konsumpcji jutro i dziś. Po lewej stronie mamy zaś krańcową stopę substytucji (iloraz krańcowych użyteczności) konsumpcji dziś konsumpcją jutrzejszą z punktu widzenia funkcji celu $\sum_{t=0}^T \beta^t u(c_t)$. Tak więc równanie Eulera można interpretować jako wyraz znanego z mikroekonomii

2 Ciągła przestrzeń stanów. Równania Eulera. nieskończony horyzont

faktu, że w optymalnym koszyku dóbr ich stopy substytucji muszą się równać ich cenom względnym.

Zauważmy jeszcze, że równanie (2.4) ma interpretację bardzo podobną do interpretacji równania Eulera. Po przeformułowaniu mamy bowiem:

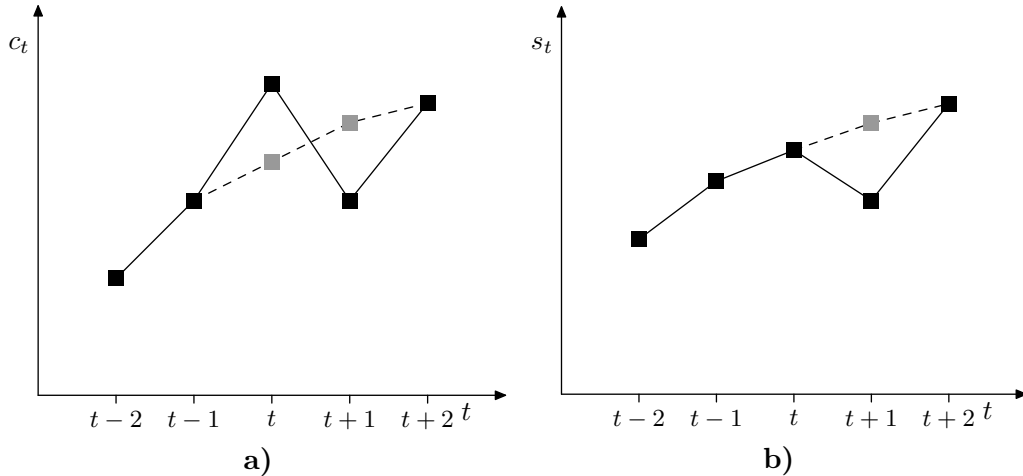
$$\frac{\beta V'_{t+1}(a_{t+1})}{u'(c_t)} = \frac{1}{(1+r)}$$

$\beta V'_{t+1}(a_{t+1})$ jest niczym innym jak krańcową użytecznością (zdyskontowaną na dziś) z jutrzejszych aktywów, zaś $\frac{1}{(1+r)}$ ich ceną w jednostkach dzisiejszej konsumpcji.

2

Ćwiczenie 2.7 Rozpatrz problem konsumenta dysponującego majątkiem w , który wydaje na konsumpcję dóbr $(x_i)_{i=0}^{N-1}$. Ceny dóbr wynoszą $(p_i)_{i=0}^{N-1}$. Konsument ma addytywną funkcję użyteczności $U(x_0, \dots, x_{N-1}) = u_0(x_0) + \dots + u_{N-1}(x_{N-1})$. Potraktuj ten problem jak problem o sekwencyjnej strukturze. Niech zmienna stanu s_i określa zasób jaki pozostał konsumentowi po wydatkach na dobra x_0, \dots, x_{i-1} mierzony w jednostkach x_i .

1. Zapisz równanie dynamiki zmiennej s_i . Czemu jest równe s_0 ?
2. Napisz równanie Bellmana.
3. Podaj warunki konieczne optymalności sterowania $(x_i)_{i=0}^{N-1}$.



Rysunek 2.1: Zaburzenie a) sterowania i b) stanu w wyprowadzeniu równania Eulera

Powyższe rozumowanie odwołujące się do cen względnych wiedzie do alternatywnego wyprowadzenia równania Eulera. Załóżmy, że $c_0, c_1, \dots, c_t, c_{t+1}, \dots, c_T$ jest optymalnym sterowaniem. Rozpatrzmy takie zaburzenie tego sterowania, by zmieniły się tylko c_t i c_{t+1} oraz a_{t+1} . Innymi słowy zaburzenie Δc_{t+1} ma kompensować wpływ zaburzenia Δc_t na $a_{t+2, t+3, \dots, T+1}$ (patrz rys. 2.1). Tą metodą uzyskamy cenę względną c_{t+1} i c_t . Ponieważ $a_{t+2} = (a_{t+1} - c_{t+1})(1+r) = ((a_t - c_t)(1+r) - c_{t+1})(1+r)$, mamy:

$$\Delta a_{t+2} = -\Delta c_t(1+r)^2 - \Delta c_{t+1}(1+r)$$

2.1 Twierdzenie o obwiedni i równania Eulera

Stąd przyjmując $\Delta a_{t+2} = 0$:

$$\Delta c_{t+1} = -\Delta c_t(1+r).$$

W przybliżeniu liniowym wpływ zaburzeń Δc_t i Δc_{t+1} na funkcję celu wyraża się wzorem:

$$\beta^t u'(c_t) \Delta c_t + \beta^{t+1} u'(c_{t+1}) \Delta c_{t+1}.$$

Ostatecznie:

$$\Delta \sum_{t=0}^T \beta^t u(c_t) = [\beta^t u'(c_t) - \beta^{t+1} (1+r) u'(c_{t+1})] \Delta c_t.$$

Jeśli sterowanie wyjściowe było optymalne, to wyrażenie w nawiasie kwadratowym powinno się zerować.¹⁴ Z tego warunku uzyskujemy równanie (2.6).

Równania Eulera można bardziej elegancko wyprowadzać podaną wyżej metodą, jeśli dokona się pewnej zamiany zmiennych. Załóżmy, że dwie różne polityki $c^{(1)}$ i $c^{(2)}$ wychodzące z danego stanu s prowadzą do tego samego stanu jutro. Jest jasne, że w takim przypadku zawsze będzie wybierana polityka o wyższej wypłacie *chwilowej*, bo jutro problem będzie taki sam (a więc i taka sama funkcja wartości) niezależnie od wyboru polityki. Ta heureka pozwala nam na rozpatrywanie wyłącznie zagadnień, w których (przy zadanym dzisiejszym stanie) jutrzejszy stan określa dzisiejsze sterowanie w sposób jednoznaczny.¹⁵ Oto zmieniona postać funkcji celu:

$$U = \sum_{t=0}^T \beta^t v(s_t, s_{t+1}),$$

gdzie $v(s_t, s_{t+1}) \equiv u(c_t)$.

Wyprowadzając równanie Eulera wystarczy przyrównać liniową część przyrostu użyteczności związanego ze zmianą stanu s_{t+1} do zera. Mamy wtedy warunek pierwszego rzędu postaci:

$$\frac{\partial U}{\partial s_{t+1}} = \beta^t v_2(s_t, s_{t+1}) + \beta^{t+1} v_1(s_{t+1}, s_{t+2}) = 0.$$

Ćwiczenie 2.8 Znajdź $v(s_t, s_{t+1})$ w problemie (2.2). Wyprowadź w podany powyżej sposób równanie (2.6).

2.1.4 Metoda mnożników Lagrange'a

Metoda mnożników (czy też funkcji) Lagrange'a jest standardowym narzędziem każdego ekonomisty wykorzystywanym do rozwiązywania zagadnień optymalizacji warunkowej. Poniżej prezentujemy jej zastosowanie do rozwiązywania problemów programowania dynamicznego.

¹⁴ Jeśli $\beta^t u'(c_t) - \beta^{t+1} (1+r) u'(c_{t+1}) > 0$ ($\beta^t u'(c_t) - \beta^{t+1} (1+r) u'(c_{t+1}) < 0$) konsument może poprawić swą sytuację przerzucając konsumpcję na dziś (jutro) przyjmując $\Delta c_t > 0$ ($\Delta c_t < 0$).

¹⁵ Formalnie, w równaniu ruchu $s_{t+1} = g(s_t, c_t)$ przy ustalonym s_t c_t jest uwikłaną funkcją s_{t+1} .

2 Ciągła przestrzeń stanów. Równania Eulera. Nieskończony horyzont

W ogólności metoda funkcji Lagrange'a sprowadza się do zastąpienia problemu maksymalizacji warunkowej

$$\max_{x_1, \dots, x_N} f(x_1, \dots, x_N) \quad \text{pw.} \quad g_j(x_1, \dots, x_N) = 0 : \quad j \in \{1, \dots, M\}$$

bezw warunkową maksymalizacją funkcji \mathcal{L} określonej następująco:

$$\mathcal{L}(x_1, \dots, x_N, \lambda_1, \dots, \lambda_M) = f(x_1, \dots, x_N) + \sum_{j=1}^M \lambda_j g_j(x_1, \dots, x_N).$$

Warunki pierwszego rzędu w powyższym zagadnieniu mają postać:

$$\frac{\partial \mathcal{L}}{\partial x_i} = 0 : \quad i \in \{1, \dots, N\},$$

$$\frac{\partial \mathcal{L}}{\partial \lambda_j} = g_j(x_1, \dots, x_N) = 0 : \quad j \in \{1, \dots, M\}.$$

Jaką postać ma funkcja Lagrange'a w problemie (2.2)? Zauważmy po pierwsze, że zmienne $(a_t)_{t=1}^{T+1}$ występują w ograniczeniach (równaniu ruchu) i są, mimo że nie występują *explicite* w funkcji celu ($\sum_{t=0}^T \beta^t u(c_t)$), zmiennymi decyzyjnymi, a więc są argumentami funkcji Lagrange'a. Drugą istotną kwestią jest występowanie nierówności ($a_{T+1} \geq 0$) w warunkach ograniczających. W takiej sytuacji mamy dwie drogi: posłużyć się twierdzeniem Kuhna-Tuckera pozwalającym korzystać z funkcji Lagrange'a w problemach z nierównościami lub też odwołać się do argumentacji, która doprowadziła nas do stwierdzenia, że wiążącym ograniczeniem jest $a_{T+1} = 0$. Wybierzemy podejście drugie. Oto funkcja Lagrange'a w omawianym problemie:

$$\mathcal{L} = \sum_{t=0}^T \beta^t u(c_t) + \sum_{t=1}^{T+1} \lambda_t [(a_{t-1} - c_{t-1})(1+r) - a_t] + \mu a_{T+1}. \quad (2.7)$$

Warunki pierwszego rzędu mają postać:

$$\frac{\partial \mathcal{L}}{\partial c_t} = \beta^t u'(c_t) - \lambda_{t+1}(1+r) = 0 : \quad t \in \{0, \dots, T\}, \quad (2.8)$$

$$\frac{\partial \mathcal{L}}{\partial a_t} = -\lambda_t + (1+r)\lambda_{t+1} = 0 : \quad t \in \{1, \dots, T\}, \quad (2.9)$$

$$\frac{\partial \mathcal{L}}{\partial \lambda_t} = (a_{t-1} - c_{t-1})(1+r) - a_t = 0 : \quad t \in \{1, \dots, T+1\}, \quad (2.10)$$

$$\frac{\partial \mathcal{L}}{\partial \mu} = a_{T+1} = 0, \quad (2.11)$$

$$\frac{\partial \mathcal{L}}{\partial a_{T+1}} = -\lambda_{T+1} + \mu = 0. \quad (2.12)$$

Równania (2.8)-(2.12) stanowią układ $3T + 4$ równań z $3T + 4$ niewiadomymi w pełni określający rozwiązanie problemu. Z równań tych zupełnie łatwo wyprowadza się równanie Eulera. Z (2.8) wynika:

$$\lambda_{t+1} = \frac{\beta^t u'(c_t)}{1+r}. \quad (2.13)$$

2.2 Nieskończony horyzont. Model Ramseya

Podstawiając $\beta^{t-1}u'(c_{t-1})/(1+r)$ i $\beta^t u'(c_t)/(1+r)$ za odpowiednio λ_t i λ_{t+1} do równania (2.9) uzyskujemy:

$$\frac{\beta^{t-1}u'(c_{t-1})}{(1+r)} = \beta^t u'(c_t),$$

co po przesunięciu o jeden okres do przodu daje równanie (2.6).

Zauważmy, że z równań (2.13) i (2.4) wynika:

$$\lambda_{t+1} = \beta^{t+1}V'_{t+1}(a_{t+1}),$$

czyli:

$$\lambda_t = \beta^t V'_t(a_t). \quad (2.14)$$

Równanie (2.14) nadaje zmiennej λ interpretację podobną do tej uzyskanej w przykładzie mikroekonomicznym na początku rozdziału. λ_t jest ceną cieniem określającą wartość zasobu a_t w jednostkach użyteczności z okresu 0. Jeśli chcielibyśmy aby zmienna λ_t wyceniała zasób w jednostkach użyteczności z okresu t („na bieżąco”)¹⁶ wystarczy zapisać funkcję Lagrange’a następująco:

$$\mathcal{L} = \sum_{t=0}^T \beta^t u(c_t) + \sum_{t=1}^{T+1} \beta^t \lambda_t [(a_{t-1} - c_{t-1})(1+r) - a_t] + \beta^{T+1} \mu a_{T+1}. \quad (2.15)$$

Ćwiczenie 2.9 Wyprowadź odpowiedniki równań (2.8)-(2.12) dla funkcji Lagrange’a (2.15). Pokaż, że w tym przypadku zachodzi $\lambda_t = V'_t(a_t)$.

2.2 Nieskończony horyzont. Model Ramseya

2.2.1 Problem centralnego planisty w modelu Ramseya

Zastanówmy się teraz nad następującym problemem:

$$\max \sum_{t=0}^{\infty} \beta^t u(c_t) \quad \text{pw.} \quad c_t + i_t \leq f(k_t), \quad k_{t+1} = (1-\delta)k_t + i_t, \quad \forall_t c_t, k_t \geq 0. \quad (2.16)$$

Jest to wersja modelu Ramseya¹⁷ (problemu centralnego planisty maksymalizującego użyteczność społeczeństwa pod warunkiem ograniczeń *całej gospodarki*)¹⁸ w czasie dyskretnym. O funkcjach u i f przyjmujemy standardowe założenia (u i f

¹⁶ Czytelnik zaznajomiony z zasadą maksimum i hamiltonianami z pewnością dostrzeże analogię ze „zwykłym” hamiltonianem i hamiltonianem wartości bieżącej.

¹⁷ W oryginalnym artykule Ramseya [18] problem przedstawiony jest w wersji bez dyskontowania, gdyż autor uważał takie podejście ze strony centralnego planisty za nieetyczne. Abstrahując od kwestii etycznych należy pamiętać, że brak dyskontowania sprawia duży problem natury technicznej a to ze względu na problem zbieżności sumy (całki) w funkcji celu. Rozwiązanie problemu zbieżności sumy (całki) w określeniu funkcji celu wymaga odwoływania się do funkcji użyteczności charakteryzujących się pewnym poziomem nasycenia (ang. *bliss*). Czynnikiem dyskontujący został prowadzony do tego modelu przez Cassa [7] i Koopmansa [12], więc pełna nazwa modelu powinna brzmieć „model Ramseya-Cassa-Koopmansa”. My będziemy za współczesnymi ekonomistami używać skróconej nazwy „model Ramseya” rozumiejąc pod nią model z dyskontowaniem.

¹⁸ Wersję „zdecentralizowaną” tego modelu przedstawimy później.

2 Ciągła przestrzeń stanów. Równania Eulera. Nieskończony horyzont

są dwukrotnie różniczkowalne oraz $u', f' > 0, u'', f'' < 0, f(0) = 0$), δ to oczywiście stopa deprecjacji.

Problem (2.16) różni się od rozpatrywanych do tej pory tylko jedną cechą — horyzontem czasowym. Jak się okazuje ta zmiana jest, przynajmniej z punktu widzenia metod rozwiązywania,¹⁹ ułatwieniem! Zacznijmy od zapisania równania Bellmana:

$$V_t(k_t) = \max_{c_t} \{u(c_t) + \beta V_{t+1}(\underbrace{(1 - \delta)k_t + f(k_t) - c_t}_{k_{t+1}})\}.$$

Zauważmy, że tak naprawdę subskrypty czasowe przy funkcji wartości są niepotrzebne — jeśli w roku 2000 mamy \bar{k} kapitału nasz problem jest taki sam jak w roku 2001 z tym samym kapitałem. Czas nie występuje *explicite* ani w funkcji produkcji, ani użyteczności, ani też w równaniu ruchu na kapitał. Innymi słowy mamy jedną funkcję wartości spełniającą równanie funkcyjne:

$$V(k_t) = \max_{c_t} \{u(c_t) + \beta V((1 - \delta)k_t + f(k_t) - c_t)\} \tag{2.17}$$

i jedną funkcję polityki:

$$c(k_t) = \arg \max_{c_t} \{u(c_t) + \beta V((1 - \delta)k_t + f(k_t) - c_t)\}. \tag{2.18}$$

Zaczniemy od wyprowadzenia równania Eulera. Z warunku pierwszego rzędu dla prawej strony równania (2.17) mamy:

$$u'(c_t) - \beta V'(k_{t+1}) = 0 \tag{2.19}$$

Zróżniczkujmy teraz równanie (2.17) obustronnie po k_t (korzystamy z twierdzenia o obwiedni):

$$V'(k_t) = \beta[1 - \delta + f'(k_t)]V'(k_{t+1}). \tag{2.20}$$

Z równania (2.19) mamy $V'(k_t) = \beta^{-1}u'(c_t)$ i $V'(k_{t+1}) = \beta^{-1}u'(c_{t+1})$. Podstawiając za $V'(k_t)$ i $V'(k_{t+1})$ w równaniu (2.20) uzyskujemy:

$$u'(c_t) = \beta[1 - \delta + f'(k_{t+1})]u'(c_{t+1}). \tag{2.21}$$

Równanie (2.21) ma taką samą interpretację jak równanie Eulera (2.6). Wyrażenie $f'(k_t) - \delta$ określa stopę zwrotu z inwestycji.

Ćwiczenie 2.10 Wzorując się na wyprowadzeniu równania Eulera ze strony 27 uzasadnij, że $1 + f'(k_t) - \delta$ jest rzeczywiście ceną względną konsumpcji w okresach t i $t + 1$.

Ćwiczenie 2.11 Funkcja Lagrange'a w modelu Ramseya ma postać:

$$\mathcal{L} = \sum_{t=0}^{\infty} \beta^t u(c_t) - \sum_{t=1}^{\infty} \beta^t \lambda_t (k_t + c_{t-1} - (1 - \delta)k_{t-1} - f(k_{t-1})).$$

Wyprowadź warunki pierwszego rzędu i pokaż, że $V'(k_t) = \lambda_t$.

¹⁹ Odpowiednie twierdzenia o istnieniu i własnościach rozwiązań będziemy udowadniać w rozdziale 3.

2.2.2 Warunek transwersalności w zagadnieniach z nieskończonym horyzontem

Rozwiązując problem alokacji zasobu w czasie odwołał się w pewnym do warunku końcowego w postaci $V_{T+1}(a_{T+1}) = 0$ lub $a_{T+1} = 0$. Warunek ten w połączeniu z równaniem Bellmana lub Eulera i warunkiem początkowym jednoznacznie określał optymalne sterowanie c_0, c_1, \dots, c_T . W modelu Ramseya nie możemy się wprost odwołać to argumentu „końca świata” by spośród wielu ścieżek $c_{0,1,\dots}$ spełniających równanie (2.21) wybrać tę optymalną. W problemach z nieskończonym horyzontem funkcję warunku końcowego pełni tzw. warunek transwersalności:

$$\lim_{t \rightarrow \infty} \beta^t V'(k_t) k_t = 0, \quad (2.22)$$

lub inaczej:

$$\lim_{t \rightarrow \infty} \beta^t \lambda_t k_t = 0, \quad (2.23)$$

który interpretuje się jako wyraz obserwacji, że przy optymalnym sterowaniu wartość (mierzona w jednostkach użyteczności) kapitału daleko w przyszłości musi zbiegać do zera. Jeśli by tak nie było przesunięcie kapitału z „nieskończoności” na dziś (większa konsumpcja dzisiaj) prowadziłyby do zwiększenia zdyskontowanego strumienia użyteczności. Rozumowanie to jest identyczne z naszym uzasadnieniem, że w modelu alokacji zasobu w czasie musi być $a_{T+1} = 0$.

Rozwiązanie szczególnego przypadku problemu Ramseya metodą równań Eulera pozostawimy jako zadanie. Poniżej zaś zaprezentujemy tzw. metodę kolejnych przybliżeń. Rozumowanie to ma dużą wartość heurystyczną. Idea, która za nim stoi, jest podstawą algorytmu rozwiązywania problemów z dyskretną przestrzenią stanów i nieskończonym horyzontem i wykorzystywana jest także z rozważaniach teoretycznych.

2.3 Metoda kolejnych przybliżeń

W problemie (2.16) przyjmijmy $\delta = 1$, $u(c) = \ln c$ i $f(k) = k^\alpha$. Równanie Bellmana ma wtedy postać:

$$V(k_t) = \max_{c_t} \{ \ln c_t + \beta V(k_t^\alpha - c_t) \}. \quad (2.24)$$

Możemy teraz zamienić zmienne podstawiając $c = k_t^\alpha - k_{t+1}$ (k_{t+1} jest nowym sterowaniem):²⁰

$$V(k_t) = \max_{k_{t+1}} \{ \ln(k_t^\alpha - k_{t+1}) + \beta V(k_{t+1}) \}. \quad (2.25)$$

Jest to równanie funkcyjne na $V(k)$. Operator po prawej stronie równania Bellmana nazwijmy T . Wtedy nasze równanie funkcyjne ma postać $V = T(V)$ (funkcja wartości jest więc punktem stałym operatora T). Metoda kolejnych przybliżeń będzie

²⁰ Taką zamianę zmiennych, która prowadzi do określenia wypłaty nie jako funkcji dzisiejszego stanu i dzisiejszego sterowania, lecz funkcji stanu dzisiejszego i jutrzejszego już pokazywaliśmy i jeszcze się z nią spotkamy. Zauważmy teraz tylko, że zamianę tę można interpretować jako takie przeformułowanie problemu, by równanie ewolucji zmiennej stanu miało postać $s_{t+1} = c_t$.

2 Ciągła przestrzeń stanów. Równania Eulera. Nieskończony horyzont

polegała na wyjściu od pewnej funkcji V_0 , policzeniu $T(V_0)$ i przyjęciu tej funkcji jako następnego przybliżenia, itd.²¹ Schemat będzie miał więc następującą postać rekurencyjną: $V_{i+1} = T(V_i)$. (Nie zapomnijmy jednak, że działamy na funkcjach nie na liczbach!)

Zaczynamy od przybliżenia $V_0(k_t) = 0$. Mamy:

$$V_1(k_t) = T(V_0)(k_t) = \max_{k_{t+1}} \{\ln(k_t^\alpha - k_{t+1})\} = \alpha \ln k_t,$$

bo optymalne $k_{t+1} = 0$.²²

Dalej:

$$V_2(k_t) = T(V_1)(k_t) = \max_{k_{t+1}} \{\ln(k_t^\alpha - k_{t+1}) + \alpha\beta \ln k_{t+1}\}.$$

Warunek pierwszego rzędu to:

$$-\frac{1}{k_t^\alpha - k_{t+1}} + \alpha\beta \frac{1}{k_{t+1}} = 0,$$

stąd:

$$k_{t+1} = \frac{\alpha\beta}{1 + \alpha\beta} k_t^\alpha.$$

Ostatecznie:

$$V_2(k_t) = \alpha\beta \ln(\alpha\beta) - (1 + \alpha\beta) \ln(1 + \alpha\beta) + \alpha(1 + \alpha\beta) \ln k_t.$$

Pełne rozwiązanie problemu zawarte jest z poniższych stwierdzeń, których dowody pozostawiamy jako ćwiczenie.

Stwierdzenie 2.1 Niech T będzie operatorem zadany wzorem:

$$T(f)(x) = \max_y \{\ln(x^\alpha - y) + \beta f(y)\} \tag{2.26}$$

i niech g_n oznacza rodzinę funkcji określonych wzorem

$$g_n = a_n + b_n \ln x \tag{2.27}$$

Wtedy $T(g_n)(x) = g_{n+1}(x)$ przy czym a_n i b_n spełniają równania rekurencyjne:

$$a_{n+1} = a_n\beta + \beta b_n \ln(\beta b_n) - (1 + b_n\beta) \ln(1 + b_n\beta), \tag{2.28}$$

$$b_{n+1} = \alpha(1 + b_n\beta). \tag{2.29}$$

Ćwiczenie 2.12 Udowodnij stwierdzenie 2.1.

Stwierdzenie 2.2 Ciągi a_n i b_n określone jak wyżej zbiegają i mają granice:

$$\lim_{n \rightarrow \infty} a_n = \frac{\ln(1 - \alpha\beta)}{1 - \beta} + \frac{\alpha\beta \ln(\alpha\beta)}{(1 - \beta)(1 - \alpha\beta)}, \tag{2.30}$$

$$\lim_{n \rightarrow \infty} b_n = \frac{\alpha}{1 - \alpha\beta}. \tag{2.31}$$

²¹ Metoda kolejnych przybliżeń pochodzi od Picarda, który stosował ją do rozwiązywania i udowodniania istnienia rozwiązań równań różniczkowych.

²² Kapitał jutrzejszy nie może być ujemny.

2.4 Algorytm kolejnych przybliżeń w zagadnieniach z dyskretną przestrzenią stanów

Ćwiczenie 2.13 Udowodnij stwierdzenie 2.2. **Wskazówka:** zacznij od ciągu b_n .

Z powyższych stwierdzeń wynika, że rozwiązaniem równania Bellmana jest funkcja wartości:

$$V(k_t) = \frac{\ln(1 - \alpha\beta)}{1 - \beta} + \frac{\alpha\beta \ln(\alpha\beta)}{(1 - \beta)(1 - \alpha\beta)} + \frac{\alpha}{1 - \alpha\beta} \ln k_t, \quad (2.32)$$

której odpowiada sterowanie:

$$k_{t+1}(k_t) = \alpha\beta k_t^\alpha,$$

czyli:

$$c(k_t) = (1 - \alpha\beta)k_t^\alpha. \quad (2.33)$$

Zauważmy, że w problemie Ramseya z logarytmiczną użytecznością, funkcją produkcji Cobba–Douglasa i pełną deprecjacją optymalne rozwiązanie charakteryzuje się stałą stopą oszczędności.

Ćwiczenie 2.14 Znajdź stan ustalony tego modelu, czyli takie k^* , dla którego jeśli $k_t = k^*$, to $k_{t+1} = k^*$. Pokaż, że dla dodatniego kapitału początkowego gospodarka zbiega do stanu ustalonego.

2.4 Algorytm kolejnych przybliżeń w zagadnieniach z dyskretną przestrzenią stanów

2.4.1 Algorytm

Istotę metody kolejnych przybliżeń nakreśliliśmy w poprzednim podrozdziale. Poniżej podajemy bazujący na niej algorytm rozwiązywania problemów z dyskretną przestrzenią stanów.

Jak w przypadku algorytmu rekursywnego z poprzedniego rozdziału, wejściem będzie macierz R $n \times m$ wypłat przypisującą każdemu stanowi i każdemu sterowaniu liczbę $R_{i,j} = u(c_j, s_i)$ oraz macierz TR $n \times m$ przejścia opisująca w jakim stanie się znajdziemy, jeśli teraz ze stanu i wybierzemy sterowanie j . Potrzebna będzie także informacja o czynniku dyskontującym β .

Ze względu fakt, iż rozpatrujemy problem z nieskończonym horyzontem, nie potrzebujemy informacji o wypłatach końcowych, a funkcje wartości i polityki stają się wektorami, w których i -ta pozycja opisuje odpowiednio funkcję wartości i optymalne sterowanie z i -tego stanu.

Metoda kolejnych przybliżeń jest przykładem tzw. algorytmu iteracyjnego. Wszystkie tego typu algorytmy mają dwie własności: wyniki (po skończonej liczbie iteracji) są prawie na pewno niedokładne i potrzebne jest pewne kryterium stopu. Standardowo za kryterium zatrzymania przyjmuje się odpowiednio małą (mniejszą od pewnej dodatniej liczby ε) normę (patrz dodatek C) poprawki wynikającej z ostatniej iteracji.

Nasz algorytm będzie wykorzystywał dwa pomocnicze wektory: wektor p wartości polityk i wektor $V^{(old)}$, w którym będziemy przechowywać funkcję wartości z poprzedniej iteracji (potrzebną do obliczenia wielkości poprawki).

Oto pełny algorytm:

2 Ciągła przestrzeń stanów. Równania Eulera. Nieskończony horyzont

1. Wczytaj β oraz macierze R i TR . Ustal ε .
2. Inicjalizuj wektory C i V .
3. Przypisz $V^{(old)} = V$.
4. Ustal stan $s = 1$.
5. Ustal sterowanie $c = 1$.
6. Przypisz $p_c = R_{s,c} + \beta V_{TR_{s,c}}^{(old)}$
7. Jeśli $c < m$, zwiększ c o jeden i idź do 6.
8. Przypisz $V_s = \max_i \{p_i\}$ i $C_s = \arg \max_i \{p_i\}$.
9. Jeśli $s < n$, zwiększ s o jeden i idź do 5.
10. Jeśli $\|V - V^{(old)}\| > \varepsilon$, idź do 3.

2.4.2 Implementacja w języku Matlab/Octave

Wszystkie oznaczenia przyjmujemy jak poprzednio. Wektor $V^{(old)}$ oznaczamy w kodzie przez `valueold`. Poziom tolerancji ε oznaczamy `tol` i przyjmujemy jego wartość równą pierwiastkowi z dokładności liczby zmiennoprzecinkowej w naszym systemie (wbudowana zmienna `eps`).

Oto kod naszego programu:²³

Plik `dpsa2.m`

```
function [C, V]=dpsa2(discount, reward, transition)
%function [C, V]=dpsa2(discount, reward, transition)
%
%dpsa2 rozwiązuje problem programowania dynamicznego
%metoda kolejnych przyblizen
%
%discount - współczynnik dyskontujacy
%reward - macierz wyplat
%transistion - macierz przejścia
%
%(c) Grzegorz Klima 2004

tol=sqrt(eps);
[n, m]=size(reward);
control=zeros(1, n);
value=zeros(1, n);
policy_values=zeros(1,m);
normdeltaval=1;

while (normdeltaval>tol)
```

²³ Nasz algorytm podobnie jak ten z poprzedniego rozdziału działa w trzech pętlach i jest podobnie wolny.

Zadania

```

valueold=value;
for i=1:n
    for j=1:m
        policy_values(j)=reward(i,j)+discount*valueold(transition(i,j));
    end
    [value(i), control(i)]=max(policy_values);
end
normdeltaval=norm(value-valueold);
end

C=control;
V=value;

```

Koniec pliku *dpsa2.m*

2

Do symulacji modeli z nieskończonym horyzontem musimy nieco zmienić poprzednio używaną funkcję `sim1`. Jako wejście do tej zmienionej funkcji nazwanej `sim2` musimy podać obok stanu początkowego, macierzy przejścia i wektora optymalnych sterowań także horyzont czasowy naszej symulacji. Poniżej podajemy kod funkcji `sim2`:

Plik *sim2.m*

```

function ret=sim2(x0, time, transition, control)
%function ret=sim2(x0, time, transition, control)
%
%sim2 symuluje ewolucje zmiennej stanu
%
%x0 - stan poczatkowy
%time - horyzont symulacji
%transition - macierz przejścia
%control - wektor sterowan
%
%(c) Grzegorz Klima 2004

x=zeros(1,time+1);
x(1)=x0;

for t=1:time
    x(t+1)=transition(x(t), control(x(t)));
end

ret=x;

```

Koniec pliku *sim2.m*

Ćwiczenie 2.15 Rozwiąż na komputerze wersję problemu optymalnej eksploatacji zasobu naturalnego z poprzedniego rozdziału z nieskończonym horyzontem. Dla stóp procentowych $r = 1\%, 2\%, \dots, 20\%$ określ czas, po którym złożo zostanie wyeksploatowane i zrób wykres tej zależności. Wyjaśnij intuicyjnie uzyskany wynik.

Zadania

Zadanie 2.1 Rozwiąż model Ramseya z ograniczonym horyzontem czasowym metodą równań Eulera. Przyjmij $\delta = 1$, $u(c) = \ln c$ i $f(k) = k^\alpha$. Problem jest więc następujący:

$$\max \sum_{t=0}^T \beta^t u(c_t) \quad \text{pw.} \quad c_t + i_t \leq k_t^\alpha, \quad k_{t+1} = i_t.$$

Wskazówka: w równaniu Eulera dokonaj podstawienia $c_t = k_t^\alpha - k_{t+1}$, a następnie $z_t = k_t/k_{t-1}^\alpha$. Znajdź i rozwiąż rekurencyjne równanie na z_t . Zauważ, że $z_{T+1} = 0$. Pokaż, że dla $T \rightarrow \infty$ funkcja polityki $k_{t+1}(k_t)$ zbiega do funkcji polityki z problemu z nieskończonym horyzontem.

Zadanie 2.2 Dokonaj zamiany zmiennych w modelu Ramseya (2.16), tak by zmienną sterującą była nie konsumpcja, lecz stopa oszczędności $s_t = i_t/f(k_t)$.

1. Zapisz równanie Bellmana.
2. Wyprowadź równanie Eulera.
3. Przyjmij jak w zadaniu poprzednim $\delta = 1$, $u(c) = \ln c$ i $f(k) = k^\alpha$. Znajdź zależność $s_{t+1}(s_t)$.
4. Pokaż, że jeśli $s_0 \neq \alpha\beta$ wtedy $s_t \rightarrow 1$. Pokaż, że wtedy nie jest spełniony warunek transwersalności, a więc optymalnym sterowaniem jest $s_t = \alpha\beta$.

Zadanie 2.3 Rozwiąż numerycznie wersję modelu Ramseya z logarytmiczną użytecznością, funkcją produkcji Cobba-Douglassa i pełną deprecjacją, korzystając z funkcji `dpsa2.m`. Przyjmij $\alpha = 0,333$ i $\beta = 0,9$. Znajdź takie k^* , dla którego $k_{t+1}(k^*) = k^*$ i dokonaj dyskretyzacji przestrzeni stanu (poziomów kapitału) na przedziale $k_{\min} = [0,1k^*, k_{\max} = 1,3k^*]$. Napisz najpierw funkcje odpowiednio wypełniające macierze `transition` i `reward` w zależności od k_{\min} , k_{\max} i gęstości podziału (liczby pośrednich poziomów kapitału). Porównaj uzyskaną funkcję polityki z wynikiem teoretycznym.

Zadanie 2.4 Rozparz problem przedsiębiorstwa optymalizującego swoje wydatki inwestycyjne w obecności kosztów instalacji:

$$\max \sum_{t=0}^{\infty} \frac{f(k_t) - i_t - \Phi\left(\frac{i_t}{k_t}\right)}{(1+r)^t} \quad \text{pw.} \quad k_{t+1} = (1-\delta)k_t + i_t, \quad \forall_t k_t, i_t \geq 0,$$

przy czym $\Phi(x) = Cx^2$, $C \geq 0$.

1. Zapisz równanie Bellmana i korzystając z twierdzenia o obwiedni wyprowadź z niego równanie Eulera.
2. Zapisz funkcję Lagrange'a i znajdź warunki pierwszego rzędu.
3. Wyprowadź równanie Eulera bezpośrednio.
4. Jaką interpretację ma tutaj mnożnik Lagrange'a (jest to tzw. q -Tobina)? Jak C wpływa na jego wielkość?

Rozdział 3

Warunki konieczne i dostateczne optymalności sterowania

3

W niniejszym rozdziale przedstawiamy podstawowe wyniki teoretyczne dotyczące zagadnienia programowania dynamicznego z nieskończonym horyzontem. Prezentacja bazuje na podręczniku Stokey, Lucasa i Prescottta [13]. Spośród wielu wyników wybrano tylko te najogólniejsze i niewymagające specjalnych wstępów teoretycznych. Czytelnika zainteresowanego szczególnymi przypadkami, a także dowodami różniczkowalności odsyłamy do tej świetnej monografii.

3.1 Definicje i założenia

W rozdziale pierwszym i drugim rozpatrując problemy optymalizacji pisaliśmy $\max_{x \in X} \{f(x)\}$ zakładając istnienie pewnego $x' \in X$ spełniającego $\forall_{x \in X} f(x) \leq f(x')$. W tym momencie rezygnujemy z tego założenia i zaczniemy nasze rozważania od supremum funkcji celu, które definiujemy poniżej.

Definicja 3.1 Przez supremum ($\sup_{x \in X} \{f(x)\}$) funkcji f określonej na X i przyjmującej wartości w $\overline{\mathbb{R}} \equiv \mathbb{R} \cup \{-\infty, +\infty\}$ rozumiemy takie $M \in \mathbb{R}$, dla którego:

$$\forall_{x \in X} f(x) \leq M \wedge \forall_{\varepsilon > 0} \exists_{x \in X} f(x) + \varepsilon \geq M, \quad (3.1)$$

lub $+\infty$, jeśli $\forall_{M \in \mathbb{R}} \exists_{x \in X} f(x) > M$, lub $-\infty$, jeśli $\forall_{x \in X} f(x) = -\infty$.

Supremum funkcji f na zbiorze X nazywamy także kresem górnym zbioru wartości funkcji (przeciwdziedziny) $f(X) \equiv \{f(x) : x \in X\}$.

Ćwiczenie 3.1 Udowodnij odpowiedniki nierówności (1.2), (1.6), (1.8) i (1.10) dla supremum.

Na potrzeby poniższych rozważań przyjmiemy, podobnie jak to czyniliśmy w rozdziale drugim, wypłatę chwilową jako funkcję dzisiejszego i jutrzejszego stanu. Oto

3 Warunki konieczne i dostateczne optymalności sterowania

formuła funkcji celu, która będzie nas obowiązywała do końca tego rozdziału (zakładamy $\beta \in (0, 1)$):

$$\sum_{t=0}^{\infty} \beta^t u(s_t, s_{t+1})$$

Po dokonaniu powyższej „zamiany zmiennych” przez politykę będziemy rozumieć decyzję, w którym stanie chcemy się znaleźć jutro. Ograniczenia polityki, to ograniczenia stanów, do których możemy przejść będąc dziś w stanie s . Oznaczmy przez S przestrzeń stanów.

Definicja 3.2 Przez $\Gamma(s_t) \subset S$ rozumiemy zbiór stanów, do których można przejść ze stanu s_t

Ćwiczenie 3.2 Określ $\Gamma(k_t)$ w modelu Ramseya (zagadnieniu centralnego planisty). Dana jest funkcja produkcji $f(k)$ i stopa deprecjacji δ .

Definicja 3.3 Ciąg $\tilde{s} = (s_t)_{t=0}^{\infty}$ nazywamy planem dopuszczalnym rozpoczynającym się w s_0 i zapisujemy $\tilde{s} \in \Pi(s_0)$, gdy $\forall_t s_{t+1} \in \Gamma(s_t)$.

Jeśli założymy, że:

$$\forall_{s \in S} \Gamma(s) \neq \emptyset, \quad (3.2)$$

wtedy wiemy, że z każdego stanu można wybrać przynajmniej jeden plan dopuszczalny.

Dodatkowo zakładamy, że $\forall_{s_0 \in S} \forall_{\tilde{s} \in \Pi(s_0)}$ istnieje (choć może być to $+\infty$ lub $-\infty$) granica:

$$\lim_{T \rightarrow \infty} \sum_{t=0}^T \beta^t u(s_t, s_{t+1}). \quad (3.3)$$

Powyższe założenia będą obowiązywały nas do końca rozdziału.

Wprowadzimy jeszcze trzy oznaczenia (\tilde{s} jest pewnym planem dopuszczalnym):

$$U_T(\tilde{s}) \equiv \sum_{t=0}^T \beta^t u(s_t, s_{t+1}) \quad (3.4)$$

$$U(\tilde{s}) \equiv \lim_{T \rightarrow \infty} U_T(\tilde{s}) \quad (3.5)$$

$$V^*(s_0) \equiv \sup_{\tilde{s} \in \Pi(s_0)} U(\tilde{s}) \quad (3.6)$$

Tak więc $U(\tilde{s})$ jest wypłatą z dopuszczalnego planu, zaś $V^*(s_0)$ to supremum wypłat z planów dopuszczalnych zaczynających się w s_0 . Jest to prawie funkcja wartości. „Prawie”, bo nie dyskutujemy jeszcze istnienia sterowania, dla którego $V^*(s_0)$ jest osiągalne.

Zauważmy, że istnienie granicy (3.3) oznacza tak naprawdę istnienie $U(\tilde{s})$. Z tego założenia wynika też następujący lemat.

Lemat 3.1 Niech \tilde{s} oznacza pewien plan dopuszczalny rozpoczynający się w s_0 , zaś \tilde{s}' oznacza część tego planu zaczynającą się od s_1 . Wtedy:

$$U(\tilde{s}) = u(s_0, s_1) + \beta U(\tilde{s}') \quad (3.7)$$

Dowód.

$$\begin{aligned} U(\tilde{s}) &= \lim_{T \rightarrow \infty} \sum_{t=0}^T \beta^t u(s_t, s_{t+1}) = \\ u(s_0, s_1) + \beta \lim_{T \rightarrow \infty} \sum_{t=0}^T \beta^t u(s_{t+1}, s_{t+2}) &= \\ u(s_0, s_1) + \beta U(\tilde{s}') &\blacksquare \end{aligned}$$

3.2 Od optymalnej polityki do równania Bellmana

Twierdzenie 3.1 Niech będzie spełnione założenie (3.2) oraz dla każdego stanu $s \in S$ i rozpoczynającego się w nim planu dopuszczalnego istnieje granica (3.3). Wtedy kres górny $V^*(s)$ zdefiniowany w (3.6) spełnia równanie funkcyjne:

$$V(s_t) = \sup_{s_{t+1} \in \Gamma(s_t)} \{u(s_t, s_{t+1}) + \beta V(s_{t+1})\} \quad (3.8)$$

Dowód. Ograniczamy się do przypadku, w którym kres górny $V^*(s)$ jest skończony.¹ Z definicji supremum mamy:

$$\forall \tilde{s} \in \Pi(s_0) V^*(s_0) \geq U(\tilde{s}) \quad (3.9)$$

oraz

$$\forall \varepsilon > 0 \exists \tilde{s} \in \Pi(s_0) V^*(s_0) \leq U(\tilde{s}) + \varepsilon \quad (3.10)$$

Ustalmy $\varepsilon > 0$ i $s_1 \in \Gamma(s_0)$. Niech $\tilde{s}' = (s_1, s_2, \dots)$ będzie pewnym planem dopuszczalnym wychodzącym z s_1 , dla którego:

$$U(\tilde{s}') \geq V^*(s_1) - \varepsilon. \quad (3.11)$$

Plan $\tilde{s} = (s_0, s_1, s_2, \dots)$ jest oczywiście dopuszczalnym planem rozpoczynającym się w s_0 . Na mocy kolejno (3.9), (3.11) i lematu 3.1 mamy:

$$V^*(s_0) \geq U(\tilde{s}) = u(s_0, s_1) + \beta U(\tilde{s}') \geq u(s_0, s_1) + \beta V^*(s_1) - \beta \varepsilon.$$

Ponieważ stan s_1 i liczba ε były wybrane arbitralnie mamy:²

$$\forall s_1 \in \Gamma(s_0) V^*(s_0) \geq u(s_0, s_1) + \beta V^*(s_1). \quad (3.12)$$

Z drugiej strony, na mocy (3.10), dla pewnego ustalonego $\varepsilon > 0$ możemy wybrać taki dopuszczalny plan $\tilde{s} \in \Pi(s_0)$, że:

$$V^*(s_0) \leq U(\tilde{s}) + \varepsilon$$

¹ Tylko takie zagadnienia są interesujące dla ekonomisty. Twierdzenie pozostaje słuszne także dla $V^*(s) = \pm\infty$. Dowód w tym przypadku pozostawiamy jako ćwiczenie.

² Jeśli $\forall \varepsilon > 0 a \geq b - \varepsilon$, to $a \geq b$. Jeśli bowiem byłoby $a < b$, wówczas wystarczyłoby wziąć $\varepsilon = (b - a)/2$ i mielibyśmy $a < b - \varepsilon$.

3 Warunki konieczne i dostateczne optymalności sterowania

Lemat 3.1 i własność (3.9) zastosowane do stanów s_0 i s_1 odpowiednio dają nam (\tilde{s}' to część planu \tilde{s} zaczynająca się „jutro”, czyli od s_1):

$$V^*(s_0) \leq U(\tilde{s}) + \varepsilon = u(s_0, s_1) + \beta U(\tilde{s}') + \varepsilon \leq u(s_0, s_1) + \beta V^*(s_1) + \varepsilon.$$

Parametr ε był wybrany dowolnie, więc:

$$\forall \varepsilon > 0 \exists s_1 \in \Gamma(s_0) V^*(s_0) \leq u(s_0, s_1) + \beta V^*(s_1) + \varepsilon. \quad (3.13)$$

Konfrontując (3.12) i (3.13) z definicją 3.1 uzyskujemy tezę. ■

Ćwiczenie 3.3 Udowodnij twierdzenie 3.1 w przypadku $V^*(s) = \pm\infty$.

Jak interpretować twierdzenie 3.1? Spełnienie równania (3.8) jest warunkiem koniecznym, by kres $V^*(s)$ był funkcją określającą supremum wypłat z dopuszczalnych sterowań wychodzących ze stanu s . Jeśli jest tylko jedna funkcja spełniająca równanie (3.8) to jest ona, jak ją nazwaliśmy, ową „prawie” funkcją wartości.

Jest zupełnie jasne, że wszystkie zagadnienia interesujące ekonomistę (modele alokacji dochodu w czasie, optymalna polityka w stosunku do zasobów naturalnych, czy też różne warianty modelu Ramseya) spełniają pierwsze z założeń naszego twierdzenia – o dostępności przynajmniej jednego sterowania prowadzącego z każdego stanu, czyli o braku „ślepej uliczki”.³ Założenie o istnieniu granicy (3.3) jest mniej oczywiste i jego spełnienie istotnie zależy od konstrukcji modelu.

Ćwiczenie 3.4 Pokaż, że gdy funkcja użyteczności chwilowej $u(s_t, s_{t+1})$ jest ograniczona choć z jednej strony (z dołu lub z góry), to granica (3.3) istnieje.

Często wykorzystywana logarytmiczna funkcja użyteczności nie jest ograniczona ani z dołu, ani z góry. Można więc wyobrazić sobie technologię dającą odpowiednio dużą stopę zwrotu i takie bardzo zmienne sterowanie, dla którego $\lim_{T \rightarrow \infty} \sum_{t=0}^T \beta^t u(s_t, s_{t+1})$ nie istnieje. Ze względu na występowanie współczynnika dyskontującego w naszym szeregu, sytuacje, w których jest on rozbieżny (nie zbiega do liczby rzeczywistej lub $\pm\infty$) ograniczają się do takich, gdzie użyteczność $u(s_t, s_{t+1})$ może jednocześnie rosnać i maleć w tempie nie mniejszym niż β .⁴ Brak ograniczenia funkcji użyteczności z dołu pozwala na arbitralnie szybki spadek użyteczności chwilowej $u(s_t, s_{t+1})$ – możemy na przykład konsumować coraz mniej (zbiegać z konsumpcją do zera tak, by $\ln c_t \rightarrow -\infty$ odpowiednio szybko). Na szczęście nie jest możliwy zbyt szybki wzrost $u(s_t, s_{t+1})$. Jest to inherentna cecha wszystkich modeli ekonomicznych – rozpatrujemy kwestie ograniczonych zasobów i – co za tym idzie – ograniczonych wypłat (mierzonych użytecznością). Jeśli istniałoby sterowanie (plan) dopuszczalny, dla którego $\lim_{T \rightarrow \infty} \sum_{t=0}^T \beta^t u(s_t, s_{t+1})$ nie istnieje, to istniałoby też sterowanie pozwalające osiągnąć użyteczność równą $+\infty$, to zaś oznaczałoby, że problem jest nieinteresujący z punktu widzenia ekonomii. Nasz argument można sformalizować. Zachodzi następujący fakt.

³ Możliwa sytuacja, w której z pewnego stanu wychodzi tylko jedna droga, np. wyczerpiemy zasób, lub startujemy z zerowym poziomem kapitału (tzw. pułapka biedy).

⁴ Wystarczy skorzystać z kryterium d’Alemberta zbieżności szeregu.

3.2 Od optymalnej polityki do równania Bellmana

Fakt 3.1 Przyjmijmy oznaczenia:

$$u^+(s_t, s_{t+1}) \equiv \max\{0, u(s_t, s_{t+1})\}$$

i

$$u^-(s_t, s_{t+1}) \equiv \max\{0, -u(s_t, s_{t+1})\}.$$

Warunkiem dostatecznym zbieżności szeregu:

$$\lim_{T \rightarrow \infty} \sum_{t=0}^T \beta^t u(s_t, s_{t+1})$$

jest spełnienie przynajmniej jednej z dwóch nierówności:⁵

$$\lim_{T \rightarrow \infty} \sum_{t=0}^T \beta^t u^+(s_t, s_{t+1}) < +\infty, \quad (3.14)$$

$$\lim_{T \rightarrow \infty} \sum_{t=0}^T \beta^t u^-(s_t, s_{t+1}) < +\infty. \quad (3.15)$$

Ćwiczenie 3.5 Udowodnij fakt 3.1.

Interesująca dla nas jest nierówność (3.14). Pokażemy, że jest ona spełniona w modelu Ramseya z pełną deprecjacją i logarytmiczną funkcją użyteczności i technologią Cobba–Douglasa. Z definicji $u(k_t, k_{t+1}) = \ln(k_t^\alpha - k_{t+1})$, mamy:

$$u(k_t, k_{t+1}) \leq \ln k_t^\alpha = \alpha \ln k_t.$$

Jutrzejszy kapitał, ze względu na technologię, musi spełniać nierówność:

$$k_{t+1} \leq k_t^\alpha.$$

Stąd:

$$k_t \leq k_0^{\alpha^t}$$

oraz

$$u(k_t, k_{t+1}) \leq \alpha \ln k_t = \alpha^{t+1} \ln k_0.$$

Ostatecznie mamy:

$$\lim_{T \rightarrow \infty} \sum_{t=0}^T \beta^t u^+(s_t, s_{t+1}) \leq \lim_{T \rightarrow \infty} \sum_{t=0}^T (\alpha\beta)^t \alpha |\ln k_0| < +\infty.$$

Sformułujemy teraz twierdzenie mówiące o podstawowej własności planów optymalnych. Definicja optymalnego planu jest naturalna.

Definicja 3.4 Plan $\tilde{s}^* \in \Pi(s_0)$ nazywamy optymalnym jeśli $U(\tilde{s}^*) = V^*(s_0)$.

Zachodzi następujące twierdzenie.

⁵ Nie jest to warunek konieczny. Aby się o tym przekonać, wystarczy wziąć np. szereg $\sum_{n=1}^{\infty} \frac{(-1)^n}{n}$.

3 Warunki konieczne i dostateczne optymalności sterowania

Twierdzenie 3.2 Niech $\tilde{s}^* = (s_t^*)_{t=0}^\infty$ będzie planem optymalnym. Wtedy:

$$\forall_{t \in \{0,1,\dots\}} V^*(s_t^*) = u(s_t^*, s_{t+1}^*) + \beta V^*(s_{t+1}^*). \quad (3.16)$$

Dowód. Z optymalności \tilde{s}^* mamy:

$$\forall_{\tilde{s} \in \Pi(s_0)} V^*(s_0^*) = U(\tilde{s}^*) = u(s_0, s_1^*) + \beta U(\tilde{s}^*) \geq U(\tilde{s}) = u(s_0, s_1) + \beta U(\tilde{s}). \quad (3.17)$$

Weźmy dowolny plan $\tilde{s} = (s_t)_{t=0}^\infty \in \Pi(s_0)$, dla którego $s_1 = s_1^*$. Wówczas:

$$\forall_{\tilde{s}' \in \Pi(s_1^*)} U(\tilde{s}^*) \geq U(\tilde{s}')$$

Skąd wynika, że:

$$U(\tilde{s}^*) = V(s_1^*)$$

Podstawiając do (3.17) uzyskujemy tezę dla $t = 0$. Tezę dla każdego $t > 0$ otrzymujemy przez indukcję z (3.17). ■

Twierdzenia 3.2 i 3.1 mówią, iż spełnienie równania Bellmana jest warunkiem koniecznym optymalności polityki, jeśli optymalna polityka istnieje. Twierdzenia te nie rozstrzygają kwestii istnienia rozwiązania ani warunków dostatecznych optymalności jakiegokolwiek planu dopuszczalnego. W sytuacji, gdy jesteśmy w stanie wskazać, że istnieje tylko jedna funkcja spełniająca równanie Bellmana i jeden plan spełniający równanie polityki, mamy pewność, iż znaleźliśmy rozwiązanie problemu. Próba odwrócenia naszych wyników, a więc kwestia dostateczności równania Bellmana, jest trudniejsza. Poniżej przedstawiamy jego podstawową dyskusję na gruncie ogólnego zagadnienia.

3.3 Dostateczność równania Bellmana

Dostateczność równania Bellmana można wykazać w pewnych szczególnych przypadkach, w których odpowiednie założenia o funkcji użyteczności i funkcji przejścia (ukrytych w wyrażeniu $u(s_t, s_{t+1})$) gwarantują jednoznaczność rozwiązania równania funkcyjnego (3.8). W tym rozdziale ograniczymy się jednak do przypadku ogólnego,⁶ pokazując kryterium pozwalające wybrać spośród wielu rozwiązań tego równania prawdziwą funkcję wartości (ściślej: funkcję określającą supremum wypłat).

Ważną rolę w naszych rozważaniach będzie ogrywała metoda kolejnych przybliżeń wprowadzona w rozdziale drugim.⁷ Przed przystąpieniem do dowodu właściwego twierdzenia podamy pewien oczywisty fakt.

Fakt 3.2 Operator T określony wzorem ($f: X \rightarrow \mathbb{R}$, $u: X \times X \rightarrow \mathbb{R}$):

$$T(f)(x) = \sup_y \{u(x, y) + \beta f(y)\} \quad (3.18)$$

jest monotoniczny tzn. $\forall_{f_1, f_2: X \rightarrow \mathbb{R}} f_1(x) \geq f_2(x)$ implikuje $T(f_1)(x) \geq T(f_2)(x)$.

Dowód. Jeśli $f_1 \geq f_2$ to $u(x, y) + \beta f_1(y) \geq u(x, y) + \beta f_2(y)$ i wystarczy skorzystać z odpowiednika nierówności (1.8) dla supremum. ■

⁶ Zainteresowanego Czytelnika odsyłamy po raz kolejny do podręcznika Stokey, Lucasa i Prescotta [13].

⁷ Tutaj jednak przyjrzymy się bliżej początkowej funkcji wartości, od której rozpoczniemy iteracje.

3.3 Dostateczność równania Bellmana

Przydatny będzie także poniższy lemat.

Lemat 3.2 Niech V będzie rozwiązaniem równania funkcyjnego (3.8) spełniającym warunek:

$$\forall_{s_0 \in S} \forall_{(s_t)_{t=0}^{\infty} \in \Pi(s_0)} \limsup_{t \rightarrow \infty} \beta^t V(s_t) \leq 0. \quad (3.19)$$

Wtedy:

$$V \leq V^*,$$

gdzie V^* spełnia warunek (3.6).

Dowód. Ograniczymy się tylko do przypadku, gdy $V(s_0)$ jest liczbą skończoną. (Gdy $V(s_0) = -\infty$ dowód jest oczywisty. Zagadnienia z $V(s_0) = \infty$ są, jak już mówiliśmy, mało interesujące dla ekonomisty.) Ustalmy liczbę $\varepsilon > 0$. Niech dodatkowo $\delta = (1 - \beta)\varepsilon$. Z definicji supremum możemy wybrać taki stan $s_1 \in \Gamma(s_0)$, aby spełniony był warunek:

$$V(s_0) \leq u(s_0, s_1) + \beta V(s_1) + \delta$$

oraz taki stan $s_2 \in \Gamma(s_1)$ by:

$$V(s_1) \leq u(s_1, s_2) + \beta V(s_2) + \delta.$$

Z powyższych nierówności otrzymujemy:

$$V(s_0) \leq u(s_0, s_1) + \beta(u(s_1, s_2) + \beta V(s_2) + \delta) + \delta.$$

Postępując podobnie dla s_2, s_3, \dots tworzymy plan $\tilde{s} \in \Pi(s_0)$, który spełnia warunek:

$$V(s_0) \leq \sum_{\tau=0}^t \beta^\tau u(s_\tau, s_{\tau+1}) + \beta^{t+1} V(s_{t+1}) + \delta(1 + \beta + \dots + \beta^t) \leq \\ U_t(\tilde{s}) + \beta^{t+1} V(s_{t+1}) + \varepsilon.$$

Przechodząc do granicy przy $t \rightarrow \infty$ i pamiętając o założeniu (3.19) otrzymujemy:

$$V(s_0) \leq U(\tilde{s}) + \varepsilon.$$

Z definicji supremum;

$$U(\tilde{s}) \leq V^*(s_0),$$

skąd mamy:

$$V(s_0) \leq V^*(s_0) + \varepsilon.$$

Ponieważ stan s_0 i liczba ε były wybrane arbitralnie,⁸ uzyskujemy tezę. ■

Możemy teraz przejść do dowodu właściwego twierdzenia.

⁸ Dokonałszy małego uchybienia formalnego nie analizując sytuacji, gdy $V(s_0) = \infty$.

3 Warunki konieczne i dostateczne optymalności sterowania

Twierdzenie 3.3 Niech T będzie operatorem określonym wzorem (3.18), zaś V_0 pewną funkcją z S w \mathbb{R} spełniającą warunki:

$$T(V_0) \leq V_0, \quad (3.20)$$

$$\forall_{s_0 \in S} \forall_{(s_t)_{t=0}^{\infty} \in \Pi(s_0)} \limsup_{t \rightarrow \infty} \beta^t V(s_t) \leq 0, \quad (3.21)$$

$$\forall_{s_0 \in S} \forall_{(s_t)_{t=0}^{\infty} \in \Pi(s_0)} U(\tilde{s}) \leq V_0(s_0). \quad (3.22)$$

Wtedy funkcja V z S w $\overline{\mathbb{R}}$:

$$V(s) = \lim_{n \rightarrow \infty} T^n(V_0)$$

jest dobrze określona. Jeśli V jest punktem stałym operatora T (spełniającym warunek $T(V) = V$), to:

$$V = V^*.$$

Dowód. Z faktu 3.2 i założenia (3.20) wiemy, że $T^{n+1}(V_0) \leq T^n(V_0)$, co oznacza, że dla dowolnego stanu s ciąg $T^n(V_0)(s)$ jest nierosnący. Jako taki ma granicę skończoną lub dąży do $-\infty$. Tak więc V jest dobrze określona. Z założenia (3.21) i lematu 3.2 uzyskujemy:

$$V \leq V^*.$$

Założenie (3.22) gwarantuje $V^* \leq V_0$. Z monotoniczności T mamy $T(V^*) \leq T(V_0)$. Twierdzenie 3.1 mówi zaś, że $T(V^*) = V^*$. Mamy więc $V^* \leq T(V_0)$. Działając znów operatorem T na obie strony tej nierówności n razy (n dowolne) otrzymujemy:

$$V^* \leq T^n(V_0),$$

co implikuje

$$V^* \leq V.$$

Zestawiając tę nierówność z nierównością $V \leq V^*$ otrzymujemy tezę. ■

Powyższe twierdzenie pokazuje, przy jakich warunkach metoda kolejnych przybliżeń zapewnia zbieżność do „prawdziwej” funkcji wartości. Nierówność (3.20) zapewnia zbieżność obliczeń, zaś warunki (3.21)–(3.22) gwarantują, że nasza granica to rzeczywiście V^* . Założenie (3.21) prowadziło nas do nierówności $V \leq V^*$, czyli zapewniało, że granica nie jest większa od funkcji wartości, zaś nierówność (3.22) gwarantowała, że wartości początkowej funkcji V_0 nie są zbyt małe i granica spełnia nierówność $V^* \leq V$. Prezentując metodę kolejnych przybliżeń po raz pierwszy w poprzednim rozdziale, nie analizowaliśmy tych warunków. Prawidłowe z formalnego punktu widzenia rozwiązanie modelu Ramseya z pełną deprecjacją, logarytmiczną użytecznością i funkcją produkcji Cobba-Douglasa pozostawiamy jako zadanie.

Możemy teraz zastanowić się jaki jest warunek dostateczny optymalności jakiegoś planu. Poniższe twierdzenie stanowi pewne odwrócenie twierdzenia 3.2.

Twierdzenie 3.4 Niech $\tilde{s}^* \in \Pi(s_0)$ będzie planem spełniającym warunek (3.16) oraz:

$$\limsup_{t \rightarrow \infty} \beta^t V^*(s_t^*) \leq 0. \quad (3.23)$$

3.4 Dostateczność równania Eulera w połączeniu z warunkiem transwersalności

Wtedy \tilde{s}^* jest planem optymalnym.

Dowód. Z równania (3.16) mamy przez indukcję:

$$\begin{aligned} V^*(s_0^*) &= u(s_0^*, s_1^*) + \beta V^*(s_1^*) = u(s_0^*, s_1^*) + \beta u(s_1^*, s_2^*) + \beta^2 V^*(s_2^*) = \\ &\dots = U_t(\tilde{s}^*) + \beta^{t+1} V^*(s_{t+1}^*). \end{aligned}$$

Przechodząc do granicy przy $t \rightarrow \infty$ i korzystając z (3.23) uzyskujemy nierówność $V^*(s_0^*) \leq U(\tilde{s}^*)$. Z definicji $V^*(s_0^*) \geq U(\tilde{s}^*)$, co daje tezę. ■

W sytuacji, w której mamy do wyboru więcej niż jedną politykę spełniającą warunek

$$V^*(s_t^*) = u(s_t^*, s_{t+1}^*) + \beta V^*(s_{t+1}^*),$$

co oznacza, że:

$$s_{t+1}^* \in \arg \max_{s_{t+1}} \{u(s_t^*, s_{t+1}) + \beta V^*(s_{t+1})\},$$

twierdzenie 3.4 pozwala wybrać to sterowanie, które jest częścią optymalnego planu. Warunek (3.23) ma intuicyjną interpretację⁹ – zdyskontowana na dziś funkcja wartości nie może być w nieskończoności dodatnia, gdyż oznaczałoby to rezygnację z części dzisiejszej użyteczności (odłożenie jej w nieskończoność).

3.4 Dostateczność równania Eulera w połączeniu z warunkiem transwersalności

W poprzednim rozdziale w miarę dokładnie omówiliśmy wyprowadzenie i interpretację równań Eulera jako warunków pierwszego rzędu. Poniżej podajemy twierdzenie, które przy pewnych dodatkowych założeniach zapewnia dostateczność równań Eulera w połączeniu z warunkiem transwersalności. Wspomniane dodatkowe warunki sprowadzają się do zapewnienia, że zerowanie się odpowiednich pochodnych implikuje osiągnięcie ekstremum.

Wyprowadzając równania Eulera w problemie:

$$\max \sum_{t=0}^T \beta^t u(s_t, s_{t+1}), \quad \text{pw. } (s_t)_{t=0}^\infty \in \Pi(s_0)$$

stwierdziliśmy, że jeśli ciąg $(s_t^*)_{t=1}^\infty$ jest rozwiązaniem (planem optymalnym), to nie jest opłacalna zmiana żadnego ze sterowań s_t , co oznacza, że \forall_t musi być spełniona inkluzja:

$$s_{t+1}^* \in \arg \max_{s_{t+1}} \{u(s_t^*, s_{t+1}) + \beta u(s_{t+1}, s_{t+2}^*)\}.$$

Jeśli funkcja u jest różniczkowalna, to warunkiem koniecznym (nie dostatecznym) optymalności s_{t+1} jest zerowanie się pochodnej maksymalizowanego wyrażenia w nawiasie, czyli:

$$u_2(s_t^*, s_{t+1}) + \beta u_1(s_{t+1}, s_{t+2}^*) = 0.$$

⁹ *Nota bene*, bardzo podobną do interpretacji warunku transwersalności.

3 Warunki konieczne i dostateczne optymalności sterowania

Rozwiązując problemy dynamicznej optymalizacji z wykorzystaniem równań Eulera musieliśmy odwoływać się do dodatkowego warunku, który nazywaliśmy warunkiem „końca świata” lub warunkiem transwersalności, „przypinającego” optymalną ścieżkę. Można powiedzieć, że spełnienie równań Eulera zapewnia odpowiednie relacje pomiędzy sterowaniami w kolejnych okresach, zaś warunek transwersalności zapewnia, że cały plan jest optymalny.

Po tym wstępie możemy przejść do ostatniego twierdzenia niniejszego rozdziału. Podajemy je w wersji z n -wymiarową przestrzenią stanów. Zakładamy, że Czytelnik zna pojęcie wypukłości w odniesieniu do funkcji z \mathbb{R}^n w \mathbb{R} . Zapis $x \cdot y$ oznacza iloczyn skalarny, u_1 oznacza wektor pochodnych po pierwszych n argumentach, u_2 po pozostałych n .

Twierdzenie 3.5 Niech $S \subset \mathbb{R}_+^n$, $u: \{(x, y) \in S \times S : y \in \Gamma(x)\} \rightarrow \mathbb{R}$, u jest funkcją ciągłą, ograniczoną, różniczkowalną we wnętrzu dziedziny, rosnącą ze względu na pierwszych n argumentów i wklęsłą. Jeśli $\tilde{s}^* \in \Pi(s_0)$ jest takim planem dopuszczalnym, że dla każdego t s_{t+1}^* należy do wnętrza $\Gamma(s_t^*)$ i spełnione jest równanie:

$$\forall_t u_2(s_t^*, s_{t+1}^*) + \beta u_1(s_{t+1}^*, s_{t+2}^*) = 0 \quad (3.24)$$

oraz

$$\lim_{t \rightarrow \infty} \beta^t u_1(s_t^*, s_{t+1}^*) \cdot s_t^* = 0, \quad (3.25)$$

to jest planem optymalnym.

Dowód. Niech $\tilde{s} \in \Pi(s_0)$ będzie dowolnym planem dopuszczalnym. Rozpatrzmy różnicę:

$$\Delta = \sum_{t=0}^{\infty} \beta^t u(s_t^*, s_{t+1}^*) - \sum_{t=0}^{\infty} \beta^t u(s_t, s_{t+1}) = \sum_{t=0}^{\infty} \beta^t [u(s_t^*, s_{t+1}^*) - u(s_t, s_{t+1})].$$

Z wklęsłości funkcji u mamy:

$$u(s_t^*, s_{t+1}^*) - u(s_t, s_{t+1}) \geq u_1(s_t^*, s_{t+1}^*) \cdot (s_t^* - s_t) + u_2(s_t^*, s_{t+1}^*) \cdot (s_{t+1}^* - s_{t+1}),$$

skąd:

$$\Delta \geq \lim_{T \rightarrow \infty} \sum_{t=0}^T \{u_1(s_t^*, s_{t+1}^*) \cdot (s_t^* - s_t) + u_2(s_t^*, s_{t+1}^*) \cdot (s_{t+1}^* - s_{t+1})\}.$$

Po przestawieniu wyrazów otrzymujemy:

$$\Delta \geq u_1(s_0^*, s_1^*) \cdot (s_0^* - s_0) + \lim_{T \rightarrow \infty} \left\{ \sum_{t=0}^{T-1} [(u_2(s_t^*, s_{t+1}^*) + \beta u_1(s_{t+1}^*, s_{t+2}^*)) \cdot (s_{t+1}^* - s_{t+1})] + \beta^T u_2(s_T^*, s_{T+1}^*) \cdot (s_{T+1}^* - s_{T+1}) \right\}.$$

Z (3.24) i tego, iż $s_0^* = s_0$ mamy:

$$\Delta \geq \lim_{T \rightarrow \infty} \beta^T u_2(s_T^*, s_{T+1}^*) \cdot (s_{T+1}^* - s_{T+1})$$

3.5 Uwagi

Po podstawieniu $u_2(s_T^*, s_{T+1}) = -\beta u_1(s_{T+1}, s_{T+2}^*)$:

$$\Delta \geq - \lim_{T \rightarrow \infty} \beta^{T+1} u_1(s_{T+1}^*, s_{T+2}^*) \cdot (s_{T+1}^* - s_{T+1}).$$

Z faktu, że $u_1 \geq 0$ i $s_{T+1} \geq 0$, uzyskujemy:

$$\Delta \geq - \lim_{T \rightarrow \infty} \beta^{T+1} u_1(s_{T+1}^*, s_{T+2}^*) \cdot s_{T+1}^*.$$

Na mocy (3.25) prawa strona równa jest zeru. Z nierówności $\Delta \geq 0$ i tego, że plan \tilde{s} wybraliśmy w sposób dowolny, wnioskujemy, że \tilde{s}^* jest planem optymalnym. ■

Jaki jest związek warunku tranwersalności (3.25) z podanymi w rozdziale 2 warunkami (2.22) i (2.23)? Załóżmy, że funkcja wartości spełniająca równanie:

$$V(s_t) = \max_{s_{t+1} \in \Gamma(s_t)} \{u(s_t, s_{t+1}) + \beta V(s_{t+1})\}$$

jest różniczkowalna. Wtedy z twierdzenia o obwiedni mamy (s_t^i oznacza i -tą zmienną stanu w chwili t):

$$\frac{\partial}{\partial s_t^i} V(s_t) = \frac{\partial}{\partial s_t^i} u(s_t, s_{t+1}),$$

tak więc wektor pochodnych $u_1(s_t^*, s_{t+1}^*)$ jest równy gradientowi funkcji wartości.

Z przyjętych założeń mamy $\forall_i s_t^i \geq 0$ i $\forall_i \frac{\partial}{\partial s_t^i} u(s_t, s_{t+1}) \geq 0$. Ciąg

$$\beta^t u_1(s_t, s_{t+1}) \cdot s_t$$

jest zatem sumą ciągów

$$\beta^t \frac{\partial}{\partial s_t^i} u(s_t, s_{t+1}) s_t^i$$

o składnikach dodatnich, jego zbieżność do zera jest więc równoważna ze zbieżnością do zera każdego ze składników.

Przy założeniu różniczkowalności funkcji wartości możemy warunek (3.25) przepisać następująco:

$$\forall_i \lim_{t \rightarrow \infty} \beta^t \frac{\partial}{\partial s_t^i} V(s_t) s_t^i = 0 \quad (3.26)$$

lub korzystając z mnożników Lagrange'a:

$$\forall_i \lim_{t \rightarrow \infty} \beta^t \lambda_t^i s_t^i = 0, \quad (3.27)$$

gdzie $\lambda_t^i = \frac{\partial}{\partial s_t^i} V(s_t)$ jest mnożnikiem stowarzyszonym z i -tą zmienną stanu.

3.5 Uwagi

Powyżej zaprezentowaliśmy tylko niektóre, najogólniejsze wyniki teorii deterministycznego programowania dynamicznego. Większość szczególnych wyników uzyskuje się dzięki odpowiednim założeniom o przestrzeni stanów i funkcji wypłat. W dodatku C prezentujemy np. twierdzenie Blackwella i jego zastosowanie do omawianych tu zagadnień.

3 Warunki konieczne i dostateczne optymalności sterowania

Większość klasycznych przykładów ekonomicznych spełnia bardzo wiele warunków regularności i ekonomiszczy nie sprawdzają przy każdej okazji (choć może powinni), czy ich rozwiązanie spełnia założenia odpowiednich twierdzeń, postępując podobnie jak my w poprzednim rozdziale zakładając np. różniczkowalność funkcji wartości, istnienie optymalnego sterowania i jego jednoznaczność. W dalszych rozdziałach będziemy postępować podobnie, choć należy pamiętać, że rezygnacja z pewnych założeń (np. odnośnie wklęsłości funkcji wydat, czy też przyjęcie $\beta = 1$) bez przesłania się teorii może prowadzić do błędnych rezultatów lub też sytuacji, w których rozwiązanie po prostu nie istnieje.

Zadania

3

Zadanie 3.1 Pokaż, że nierówność (3.14) jest spełniona w ogólnej wersji modelu Ramseya, z dowolną stopą deprecjacji $\delta \in [0, 1)$, jeśli tylko spełnione są warunki Inady $\lim_{c \rightarrow \infty} u'(c) = 0$ i $\lim_{k \rightarrow \infty} f'(k) = 0$. **Wskazówka:** Udowodnij, że, jeśli funkcja $f(x)$ rosnąca i wklęsła spełnia warunek $\lim_{x \rightarrow \infty} f'(x) = 0$, to można ją ograniczyć z góry przez funkcję $a + bx$, dla arbitralnego $b > 0$.

Zadanie 3.2 Rozpatrz problem z podrozdziału 2.3. Pokaż, że funkcja $V^a(k) = 0$ nie spełnia warunków (3.20)-(3.22). Pokaż, że funkcja $V^b(k) = (\alpha \ln k)/(1 - \alpha\beta)$ spełnia rzeczony warunki. Pokaż, że w tym szczególnym przypadku:

$$\lim_{n \rightarrow \infty} T^n(V^b)(k) = \lim_{n \rightarrow \infty} T^n(V^a)(k) = \frac{\ln(1 - \alpha\beta)}{1 - \beta} + \frac{\alpha\beta \ln(\alpha\beta)}{(1 - \beta)(1 - \alpha\beta)} + \frac{\alpha}{1 - \alpha\beta} \ln k.$$

Rozdział 4

Interakcje optymalizujących podmiotów. Równowaga ogólna

4

W poprzednich rozdziałach zajmowaliśmy się wyłącznie teorią podejmowania decyzji, a więc szukaniem rozwiązań problemu *pojedynczego* decydenta. Oczywiście spostrzeżenie, że w gospodarce zderzają się interesy bardzo wielu podmiotów pragnących optymalizować swoje decyzje sugeruje, iż powinniśmy zmierzać do tworzenia teorii opisującej ich interakcje, a więc ostatecznie teorii równowagi. W niniejszym rozdziale zajmiemy się analizą prostych przykładów (z zakresu czystej wymiany), podobnych do tych, jakie można spotkać w podręcznikach mikroekonomii,¹ oraz pewnymi ogólnymi wynikami teorii dobrobytu odkładając konstrukcje i analizy dynamicznych modeli równowagi ogólnej na później. W poniższej prezentacji będziemy odwoływać się do wybranych rezultatów z analizy funkcjonalnej zamieszczonych w dodatku C.

Zakładamy *implicite* istnienie równowagi i koncentrujemy się na warunkach ją określających oraz jej własnościach. Dowód twierdzenia o istnieniu równowagi konkurencyjnej Czytelnik znajdzie u Arrowa i Debreu [4].

4.1 Czysta wymiana w dwóch okresach

4.1.1 Problem jednego konsumenta

Zacznijmy rozważania od analizy problemu konsumenta żyjącego w dwóch okresach ($t = 0, 1$) i czerpiącego użyteczność z konsumpcji pewnego dobra w okresie 0 (c_0) i w okresie 1 (c_1) określoną wzorem ($0 < \beta < 1$):²

$$u(c_0) + \beta u(c_1).$$

Konsument wyposażony jest początkowo w ilości y_0 i y_1 dobra w okresie pierwszym i drugim odpowiednio. Swojego wyposażenia początkowego (otrzymywanego

¹ Patrz np. Varian [23], Mas-Colell, Whinston i Green [14].

² O funkcji użyteczności chwilowej u przyjmujemy standardowe założenia.

4 Interakcje optymalizujących podmiotów. Równowaga ogólna

dobry) nie może przechowywać. W sytuacji, w której nie ma możliwości oszczędzania i pożyczania jego wybór, ile konsumować w każdym z okresów, będzie prosty: $c_0 = y_0$, $c_1 = y_1$. Sytuację tę przedstawia rysunek 4.1a. Na rysunku pokazana jest krzywa obojętności odpowiadająca użyteczności $u(y_0) + \beta u(y_1)$.

Możliwość pożyczania i oszczędzania po pewnej stopie r poprawia sytuację naszego konsumenta.³ Jego nowy wybór i odpowiednią krzywą obojętności przedstawia rysunek 4.1b.

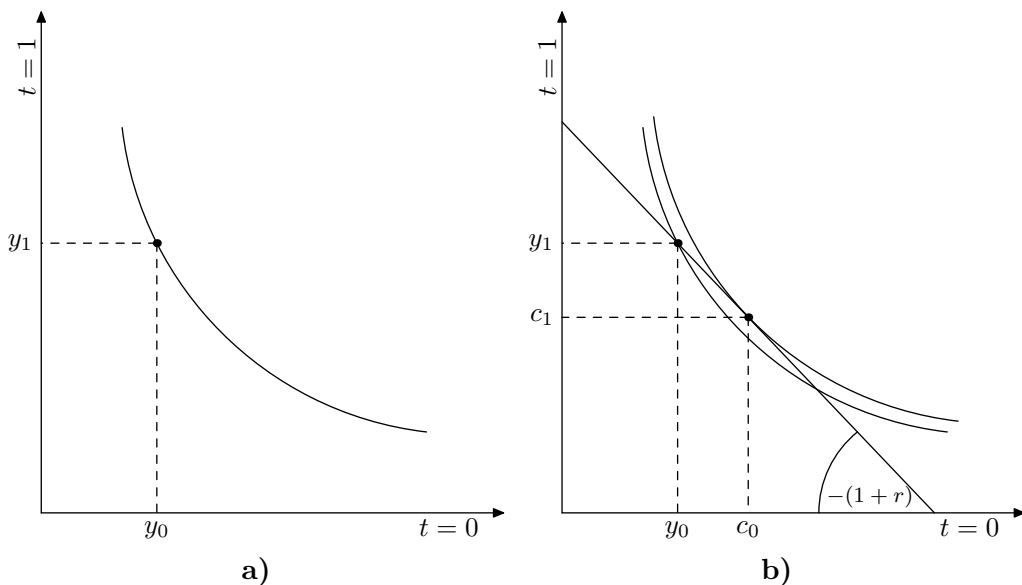
Warunek pierwszego rzędu optymalności wyboru pary (c_0, c_1) zawarty w równaniu Eulera

$$\frac{\beta u'(c_1)}{u'(c_0)} = \frac{1}{1+r} \quad (4.1)$$

jest warunkiem styczności krzywej obojętności do linii ograniczenia budżetowego $c_0 + c_1/(1+r) = y_0 + y_1/(1+r)$.

Ćwiczenie 4.1 Pokaż, że równanie (4.1) jest rzeczywiście warunkiem pierwszego rzędu w problemie:

$$\max u(c_0) + \beta u(c_1) \quad \text{pw.} \quad c_0 + \frac{c_1}{1+r} \leq y_0 + \frac{y_1}{1+r}.$$



Rysunek 4.1: Problem konsumenta żyjącego w dwóch okresach: **a)** bez dostępu do rynku kredytowego, **b)** mogącego oszczędzać i udzielać pożyczek po stopie r

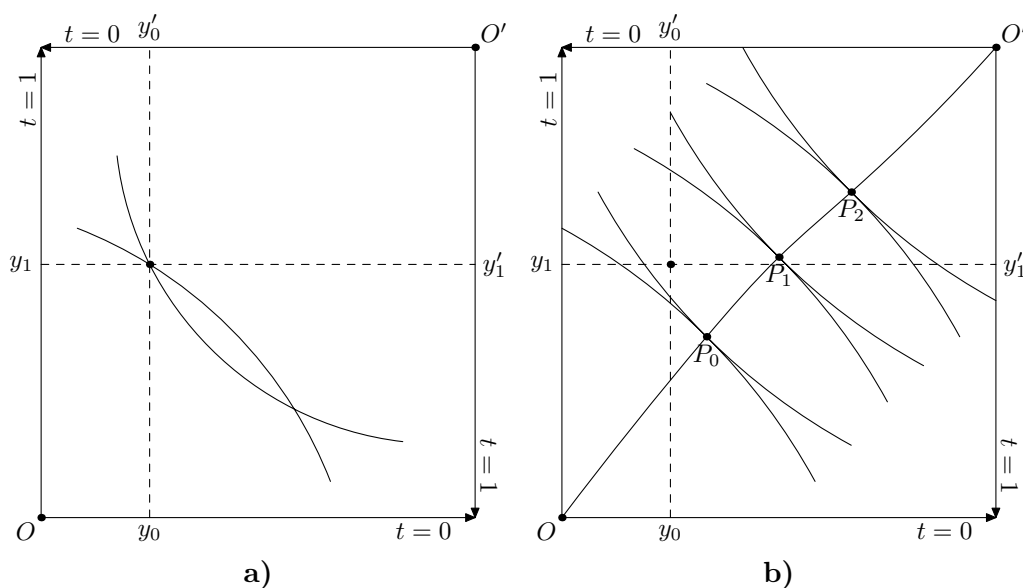
Możliwość pożyczania i oszczędzania (poprzez pożyczanie komuś) oraz stopa procentowa mierząca cenę względną konsumpcji w okresach $t = 0$ i $t = 1$ są konsekwencją interakcji podmiotów gospodarczych. Poniżej pokażemy jak tworzy się rynek kredytowy w gospodarce składającej się z dwóch podmiotów żyjących dwa okresy i dysponujących wyposażeniami początkowymi (y_0, y_1) (pierwszy) i (y'_0, y'_1) (drugi).

³ Poza przypadkiem, gdy r ma taką wartość, przy której optymalnym wyborem jest znów $c_0 = y_0$, $c_1 = y_1$.

4.1 Czysta wymiana w dwóch okresach

4.1.2 Dwóch konsumentów. Alokacje. Efektywność

Wygodnym narzędziem służącym do analizowania rynku z dwoma konsumentami i dwoma dobrami jest tzw. skrzynka (prostokąt) Edgewortha. W naszym przypadku mamy co prawda jedno dobro, ale pamiętamy, że to samo dobro w różnych momentach czasu jest czymś innym. Szerokość prostokąta to $y_0 + y'_0$ (łączna ilość dobra w okresie $t = 0$), zaś wysokość to $y_1 + y'_1$. Lewy dolny róg skrzynki jest początkiem układu współrzędnych opisującego sytuację pierwszego konsumenta, prawy górny róg to początek (odbitego względem środka) układu współrzędnych opisującego sytuację drugiego. Rysunek 4.2a przedstawia skrzynkę Edgewortha ilustrującą nasz problem oraz krzywe obojętności obu konsumentów przechodzące przez ich wyposażenia początkowe.



Rysunek 4.2: Skrzynka Edgewortha – jedno dobro, dwóch konsumentów żyjących dwa okresy: **a)** krzywe obojętności przechodzące przez wyposażenia początkowe, **b)** przykłady alokacji optymalnych w sensie Pareto

Przed przystąpieniem do dalszych rozważań podamy kilka określeń i definicji.

Niech L będzie pewną przestrzenią liniową nad ciałem liczb rzeczywistych \mathbb{R} . L interpretujemy jako przestrzeń dóbr. W naszym przykładzie jednego dobra w dwóch okresach (równoważnie: dwóch dóbr) L można utożsamić z \mathbb{R}^2 . Niech dalej $\mathcal{J} = \{1, \dots, m\}$ będzie zbiorem indeksującym konsumentów (w naszym przykładzie $\mathcal{J} = \{1, 2\}$). i -ty konsument dokonuje wyboru elementu $x^i \in X^i \subset L$ zwanego wiązką konsumpcji. Ograniczenia zbioru X^i są narzucone przez model. Jednym z najważniejszych jest zapewnienie, by konsumpcja każdego z dóbr była nieujemna. W naszym przypadku możemy przyjąć $X^i = \mathbb{R}_+^2$.

Definicja 4.1 Niech $X^1, \dots, X^m \subset L$ będą zbiorami wiązek konsumpcji. Alokacją w czystej wymianie nazywamy element (x^1, \dots, x^m) iloczynu kartezjańskiego $X^1 \times \dots \times X^m$, a więc dowolne przyporządkowanie konsumentom wiązek konsumpcji.

4 Interakcje optymalizujących podmiotów. Równowaga ogólna

Definicja 4.2 Niech $z^i \in X^i$ oznacza wyposażenie początkowe (wiązkę początkową) i -tego konsumenta. Alokację (x^1, \dots, x^m) nazwiemy dopuszczalną jeśli łączna konsumpcja każdego z dóbr nie przekracza ich łącznej ilości (połączonych wyposażzeń początkowych wszystkich uczestników wymiany). W przypadku k dóbr, L można utożsamić z \mathbb{R}^k i alokacja dopuszczalna musi spełniać:

$$\begin{aligned} \sum_i x_1^i &\leq \sum_i z_1^i, \\ \sum_i x_2^i &\leq \sum_i z_2^i, \\ &\vdots \\ \sum_i x_k^i &\leq \sum_i z_k^i. \end{aligned}$$

W rozpatrywanym przez nas przykładzie alokacją jest czwórka (c_0, c_1, c'_0, c'_1) ; alokacją dopuszczalną taka czwórka (c_0, c_1, c'_0, c'_1) , dla której $c_0 + c'_0 \leq y_0 + y'_0$ i $c_1 + c'_1 \leq y_1 + y'_1$.

Konsumenci dokonują wyborów wiązek z X^i na podstawie swoich preferencji.

Definicja 4.3 Niech L będzie pewną przestrzenią liniową, $X \subset L$ będzie zbiorem wiązek konsumpcji. Relację \preceq nazywamy relacją słabej preferencji na zbiorze X ($x \preceq y$ czytamy: y nie jest gorsze niż x), jeśli \preceq jest przechodnia ($x \preceq y \wedge y \preceq z \Rightarrow x \preceq z$), zwrotna ($x \preceq x$) i zupełna ($\forall x, y \in X (x \preceq y) \vee (y \preceq x)$).

Definicja 4.4 Niech X będzie zbiorem wiązek konsumpcji. Relację \prec nazywamy relacją mocnej preferencji (lub po prostu preferencji) na zbiorze X ($x \prec y$ czytamy: y jest lepsze niż x), jeśli $x \prec y \iff (x \preceq y) \wedge \sim (y \preceq x)$.

Funkcja użyteczności $u: X \rightarrow \mathbb{R}$ zadaje preferencje w naturalny sposób – wystarczy przyjąć $x \preceq y \iff u(x) \leq u(y)$ i $x \prec y \iff u(x) < u(y)$.

Definicja 4.5 Niech $X^i \subset L, i \in \mathcal{J}$ będą zbiorami wiązek konsumpcji, $\preceq^i, i \in \mathcal{J}$ oraz $\prec^i, i \in \mathcal{J}$ określonymi na nich relacjami słabej i mocnej preferencji. Alokację $(\bar{x}^i) \in X^1 \times \dots \times X^m$ nazywamy efektywną w sensie Pareto, jeśli jest dopuszczalna i nie istnieje taka alokacja dopuszczalna $(x^i) \in X^1 \times \dots \times X^m$, dla której $\forall i \in \mathcal{J} \bar{x}^i \preceq x^i$ i $\exists i \in \mathcal{J} \bar{x}^i \prec x^i$.

Innymi słowy alokacja jest efektywna w sensie Pareto, jeśli nie można poprawić niczyjej sytuacji nie pogarszając jednocześnie sytuacji któregośkolwiek z pozostałych podmiotów.

Spójrzmy jeszcze raz na rysunek 4.2a. Z racji samej konstrukcji skrzynki Edgewortha każdy leżący w niej pojedynczy punkt odpowiada alokacji dopuszczalnej. Przyjrzyjmy się teraz alokacji wyjściowej (y_0, y_1, y'_0, y'_1) . Obszar leżący na prawo i w górę od krzywej obojętności pierwszego konsumenta jest preferowany nad wiązkę wyjściową (y_0, y_1) , zaś obszar leżący na lewo i w dół od krzywej obojętności drugiego konsumenta jest preferowany nad wiązkę wyjściową (y'_0, y'_1) . Część wspólna tych obszarów jest niepusta, jest to obszar („soczewka”) pomiędzy krzywymi obojętności. Wynika z tego, że alokacja początkowa nie jest efektywna w sensie Pareto.

4.1 Czysta wymiana w dwóch okresach

Jak wyglądają efektywne alokacje? Krzywe obojętności przechodzące przez punkt alokacji efektywnej nie mogą przecinać się w innym punkcie obejmując niepusty obszar, w którym koszyki są ściśle preferowane, innymi słowy muszą być do siebie styczne. Trzy przykłady alokacji efektywnych w sensie Pareto przedstawia rysunek 4.2b. Zaznaczona jest także krzywa łącząca wszystkie takie alokacje. To, czy i jaka alokacja, zostanie osiągnięta zależy od sposobu wymiany oraz początkowych wyposażań. Np. drugi konsument na pewno nie zgodzi się na alokacje P_1 i P_2 (mimo tego, że są efektywne w sensie Pareto), gdyż w tych przypadkach ma na pewno mniejszą użyteczność niż w sytuacji wyjściowej (dlaczego?). Poniżej pokażemy jedną z metod dokonywania alokacji i jej wynik – równowagę konkurencyjną.

4.1.3 Równowaga konkurencyjna

Wyobraźmy sobie licytatora⁴ proponującego obu konsumentom pewną cenę względną konsumpcji w okresach $t = 0$ i $t = 1$ równą $1 + r$. Po zapoznaniu się z proponowaną ceną konsumenci wybierają poziom konsumpcji w obu okresach przy zadanych ograniczeniach budżetowych (określonych przez ich wyposażenia początkowe i r), zgłaszają swój popyt brutto, czyli wektory (c_0, c_1) i (c'_0, c'_1) oraz popyt (czy też podaż) netto, czyli ilość dobra z okresu 0, którą chcą pożyczyć z rynku (lub pożyczyć komuś) i ilość dobra z okresu 1, którą zobowiązują się oddać (której zwrotu oczekują). Popyt (podaż) netto opisują wektory $(c_0 - y_0, c_1 - y_1)$ i $(c'_0 - y'_0, c'_1 - y'_1)$. Licytator konfrontuje zgłoszenia obu konsumentów i w sytuacji, w której rynek nie jest zrównoważony ($c_0 + c'_0 \neq y_0 + y'_0$ lub $c_1 + c'_1 \neq y_1 + y'_1$) proponuje nową cenę $1 + r$. Rysunek 4.3a przedstawia optymalne wybory konsumentów przy pewnym r i krzywe obojętności im odpowiadające. Zauważmy, że (c_0, c_1, c'_0, c'_1) nie jest alokacją dopuszczalną, bo $c_1 + c'_1 > y_1 + y'_1$.

Przed przystąpieniem do poszukiwań takiego r dla którego wybory konsumentów prowadzą do alokacji dopuszczalnej i zrównoważającej rynek przedstawimy pewien ważny wynik, który nie zależy od przyjętego r .

Wybory zarówno (c_0, c_1) jak i (c'_0, c'_1) spełniają ograniczenia budżetowe (przy dowolnej stopie r):

$$c_0 + \frac{1}{1+r}c_1 = y_0 + \frac{1}{1+r}y_1,$$

$$c'_0 + \frac{1}{1+r}c'_1 = y'_0 + \frac{1}{1+r}y'_1,$$

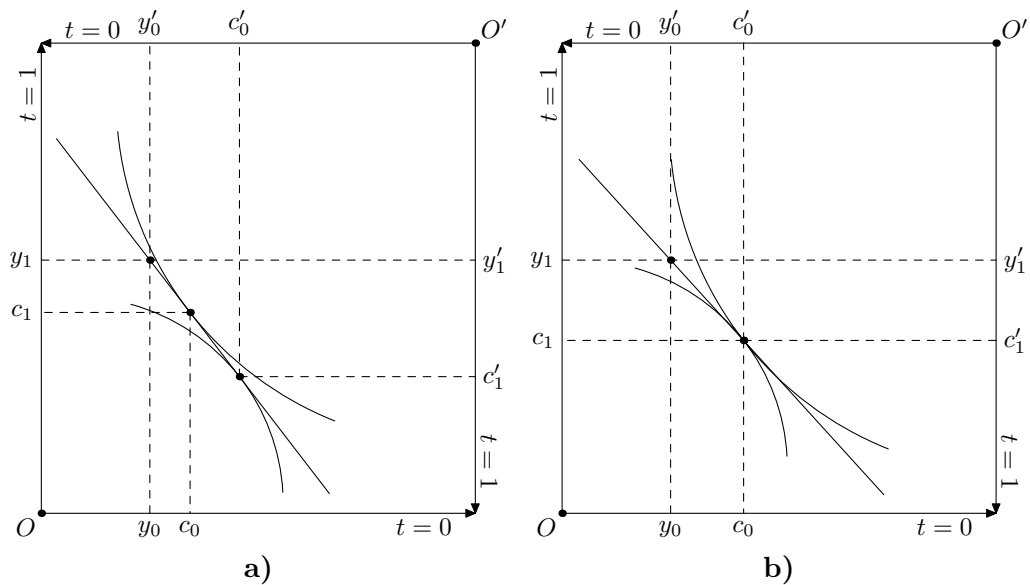
które po przeformułowaniu mają następującą postać:

$$c_0 - y_0 + \frac{1}{1+r}(c_1 - y_1) = 0,$$

$$c'_0 - y'_0 + \frac{1}{1+r}(c'_1 - y'_1) = 0.$$

⁴ Taką fikcyjną instytucję zwykło się nazywać „niewidzialną ręką rynku” lub walrasowskim licytатorem. Walrasowski licytator jest metaforą konkurencyjnego rynku z dużą liczbą sprzedających i kupujących, na którym bardzo (wręcz nieskończenie) szybko ustalają się ceny. Oczywiście analiza rynku dwóch konsumentów poprzez odwołanie się do licytatora jest pewnym nadużyciem myślowym. Omawiamy ten przypadek ze względu na jego dużą wartość dydaktyczną i łatwą graficzną interpretację wyników.

4 Interakcje optymalizujących podmiotów. Równowaga ogólna



Rysunek 4.3: Skrzyńka Edgewortha: **a)** optymalne decyzje konsumentów przy pewnej stopie procentowej r (niezapewniającej równowagi), **b)** równowaga konkurencyjna

Dodając powyższe równości stronami uzyskujemy:

$$[(c_0 + c'_0) - (y_0 + y'_0)] + \frac{1}{1+r}[(c_1 + c'_1) - (y_1 + y'_1)] = 0. \quad (4.2)$$

Powyższa równość stwierdza iż łączna wartość (wycena jest dokonywana na okres $t = 0$ według ceny względnej $1/(1+r)$) nadwyżek łącznego popytu ($c_0 + c'_0$ i $c_1 + c'_1$) nad łącznymi wyposażeniami początkowymi ($y_0 + y'_0$ i $y_1 + y'_1$) jest równa zero i to dla dowolnego r . Powyższy wynik ma charakter bardzo ogólny (bo jest wyłącznie konsekwencją dodania do siebie wszystkich ograniczeń budżetowych) i nosi nazwę prawa Walrasa.⁵

Zauważmy, że jeśli przy pewnym r zrównoważony był rynek w okresie $t = 0$

⁵ Rozpatrzmy np. przypadek n dóbr i m konsumentów. Niech $(p_i)_{i=1}^n$ będzie wektorem cen, $(\omega_i^j)_{i=1}^n$ wyposażeniem początkowym j -tego konsumenta, zaś $(x_i^j)_{i=1}^n$ wektorem jego wyborów (wiązką, którą wybiera). Ograniczenie budżetowe j -tego konsumenta ma postać:

$$\sum_{i=1}^n p_i(x_i^j - \omega_i^j) = 0.$$

Sumując ograniczenia budżetowe wszystkich konsumentów mamy:

$$\sum_{j=1}^m \sum_{i=1}^n p_i(x_i^j - \omega_i^j) = 0,$$

co po przestawieniu wyrazów daje:

$$\sum_{i=1}^n p_i \left(\sum_{j=1}^m x_i^j - \sum_{j=1}^m \omega_i^j \right) = 0.$$

4.1 Czysta wymiana w dwóch okresach

$(c_0 + c'_0 = y_0 + y'_0)$, to jednocześnie musiałyby być zrównoważony rynek w okresie $t = 1$ (i *vice versa*),⁶ bo z (4.2) mielibyśmy:

$$\underbrace{[(c_0 + c'_0) - (y_0 + y'_0)]}_{=0} + \frac{1}{1+r} [(c_1 + c'_1) - (y_1 + y'_1)] = 0,$$

co daje

$$(c_1 + c'_1) - (y_1 + y'_1) = 0.$$

Wróćmy do zagadnienia poszukiwania stopy r równoważącej rynek tworzony przez naszych dwóch konsumentów. Jakie powinna ona spełniać warunki? Przy zadanych wyposażeniach początkowych (y_0, y_1) oraz (y'_0, y'_1) r wyznacza wektory (c_0, c_1) oraz (c'_0, c'_1) poprzez układ warunków:

$$\frac{\beta u'(c_1)}{u'(c_0)} = \frac{1}{1+r}, \quad (4.3)$$

$$c_0 + \frac{c_1}{1+r} = y_0 + \frac{y_1}{1+r} \quad (4.4)$$

dla pierwszego konsumenta i:

$$\frac{\beta' u'(c'_1)}{u'(c'_0)} = \frac{1}{1+r}, \quad (4.5)$$

$$c'_0 + \frac{c'_1}{1+r} = y'_0 + \frac{y'_1}{1+r} \quad (4.6)$$

dla drugiego. Globalne ograniczenia ujęte są w równaniach:

$$(c_0 + c'_0) - (y_0 + y'_0) = 0, \quad (4.7)$$

$$(c_1 + c'_1) - (y_1 + y'_1) = 0, \quad (4.8)$$

przy czym widzimy, że na mocy prawa Walrasa równanie (4.8) jest zbędne (wynika ono z równań (4.4), (4.6) i (4.7)). Równania (4.3)-(4.7) stanowią układ 5 równań z 5-oma niewiadomymi. Jeśli układ ten ma rozwiązanie, nazywamy je równowagą konkurencyjną. Ściślej: w wymianie z n dobrami równowagą nazywamy taki wektor n cen (lub $n - 1$ cen względnych) przy których wybory (maksymalizujących swoją użyteczność) konsumentów łącznie tworzą alokację dopuszczalną. Równowagę w analizowanym przez nas przykładzie dwóch konsumentów ilustruje rysunek 4.3b.

Ćwiczenie 4.2 Przyjmij funkcje użyteczności chwilowej obu konsumentów w postaci $u(c) = \ln c$ i załóż, że $\beta = \beta'$. Wyznacz równowagę konkurencyjną przy pewnych wyposażeniach początkowych (y_0, y_1) i (y'_0, y'_1) .

Zauważmy, że równowaga ta jest efektywna w sensie Pareto, nie ma możliwości poprawy sytuacji któregoś z konsumentów bez pogarszania sytuacji drugiego. Krzywe obojętności obu konsumentów są styczne do tej samej prostej ograniczenia budżetowego, a więc do siebie. Dzieje się tak nieprzypadkowo, jest to teza pierwszego twierdzenia dobrobytu, które (w nieco ogólniejszej i bardziej abstrakcyjnej wersji) udowodnimy w dalszej części niniejszego rozdziału. Najpierw dokonamy pewnego uogólnienia naszego przykładu i przeanalizujemy dwa podejścia do równowagi konkurencyjnej w dynamicznym (uwzględniającym wpływ czasu) ujęciu.

⁶ W przypadku n dóbr zrównoważenie $n - 1$ rynków zapewnia poprzez prawo Walrasa równowagę na n -tym.

4.2 Wymiana w horyzoncie nieskończonym. Równowaga Arrowa-Debreu i rekursywna

Zanim przejdziemy do analizy równowagi w czystej wymianie z nieskończonym horyzontem Czytelnikowi należy się uwaga dotyczących używanej przez nas nomenklatury. W literaturze mikroekonomicznej elementy zbioru X^i (ang. *consumption set*) wybierane przez konsumentów określa się mianem wiązek lub koszyków dóbr (ang. *consumption bundle*), zaś wiązkę początkową z^i zwie się wyposażeniem początkowym (ang. *initial endowment*). Tego zestawu określeń używaliśmy w poprzednim podrozdziale i będziemy z niego korzystać w dyskusji twierdzeń ekonomii dobrobytu. W analizie modeli dynamicznych, którymi zajmujemy się w niniejszej książce, o wiele bardziej naturalne wydaje się korzystanie z nazewnictwa przyjętego w makroekonomii. Wyposażenie początkowe będziemy nazywać *strumieniem dochodów* $(y_t^i)_{t=0}^\infty$, zaś wiązkę konsumpcji (przedmiot decyzji konsumenta) *strumieniem konsumpcji* $(c_t^i)_{t=0}^\infty$.

4.2.1 Wymiana w okresie $t = 0$

Wyobraźmy sobie sytuację n konsumentów żyjących nieskończenie długo, których użyteczność ze strumienia konsumpcji jedynego dobra w gospodarce dana jest wzorem (i jest indeksem konsumenta):

$$\sum_{t=0}^{\infty} \beta^t u(c_t^i). \quad (4.9)$$

i -ty konsument ma zapewniony strumień dochodów $(y_t^i)_{t=0}^\infty$.

W ujęciu Arrowa-Debreu konsumenci przystępują do wymiany w okresie $t = 0$. O rynku zakładamy, że jest zupełny, co znaczy, że przedmiotem handlu może być konsumpcja w każdym momencie w przyszłości. Transakcje sprowadzające się do wymiany praw do konsumpcji w przyszłości (możemy sobie wyobrazić to jako wymianę weksli na dobro konsumpcyjne) kończą się przed przystąpieniem do konsumpcji, potem zaś uczestnicy wymiany realizują swoje prawa. Niech $(p_t)_{t=0}^\infty$ będzie (nieskończonym) wektorem cen konsumpcji (ściślej: wektorem cen praw do jednostki konsumpcji w kolejnych okresach). Ograniczenie budżetowe i -tego konsumenta można zapisać następująco:

$$\sum_{t=0}^{\infty} p_t c_t^i \leq \sum_{t=0}^{\infty} p_t y_t^i. \quad (4.10)$$

Lagranżan problemu maksymalizacji wyrażenia (4.9) pod warunkiem (4.10) ma postać:

$$\mathcal{L}^i = \sum_{t=0}^{\infty} \beta^t u(c_t^i) - \lambda^i \sum_{t=0}^{\infty} p_t (y_t^i - c_t^i). \quad (4.11)$$

Różniczkując \mathcal{L}^i po c_t^i i przyrównując wynik do zera uzyskujemy warunki pierwszego rzędu optymalności wyborów i -tego konsumenta:

$$\beta^t u'(c_t^i) = \lambda^i p_t. \quad (4.12)$$

4.2 Wymiana w horyzoncie nieskończonym. Równowaga Arrowa-Debreu i rekursywna

Przed przystąpieniem do dalszych rozważań należy się pewien komentarz. Milcząco założyliśmy zbieżność sum w nierówności (4.10). Przy założeniu, że (dodatnie) ciągi $(y_t^i)_{t=0}^{\infty}$ i $(c_t^i)_{t=0}^{\infty}$ są ograniczone, do zbieżności szeregów w (4.10) wystarcza zbieżność szeregu

$$\sum_{t=0}^{\infty} p_t. \quad (4.13)$$

Oczekiwanie, by ceny konsumpcji daleko w przyszłości malały na tyle szybko aby zapewnić zbieżność szeregu (4.13) wydaje się całkiem naturalne. Ograniczenia postawione wobec ciągu $(p_t)_{t=0}^{\infty}$ można wyobrazić sobie jako nałożenie na licytatora obowiązku proponowania wyłącznie takich cen, przy których majątki wszystkich uczestników wymiany są skończone.

Równowagę ogólną (Arrowa-Debreu) w naszym modelu definiujemy podobnie jak robiliśmy to w przypadku dwóch okresów i dwóch konsumentów. Jest to po prostu taki wektor cen $(p_t)_{t=0}^{\infty}$, dla którego wybory konsumentów (rozwiązania problemu maksymalizacji wyrażenia (4.9) pod warunkiem (4.10)) prowadzą do alokacji dopuszczalnej czyli takiej, w której dla każdego t :

$$\sum_{i=1}^n c_t^i \leq \sum_{i=1}^n y_t^i. \quad (4.14)$$

Rozpatrzmy przykład rynku składającego się z dwóch konsumentów dysponujących strumieniami dochodów $(y_t^1)_{t=0}^{\infty} = (1, 0, 1, 0, 1, 0, 1, \dots)$ i $(y_t^2)_{t=0}^{\infty} = (0, 1, 0, 1, 0, 1, 0, \dots)$. Równowagę konkurencyjną określa następujący układ równań:

$$\begin{aligned} \beta^t u'(c_t^1) &= \lambda^1 p_t, & \beta^t u'(c_t^2) &= \lambda^2 p_t, \\ \sum_{t=0}^{\infty} p_t c_t^1 &= \sum_{t=0}^{\infty} p_{2t}, & \sum_{t=0}^{\infty} p_t c_t^2 &= \sum_{t=0}^{\infty} p_{2t+1}, \\ c_t^1 + c_t^2 &= 1. \end{aligned}$$

Zauważmy, że globalne ilości dobra są stałe w czasie (równe 1). Pozwala to przypuszczać, iż konsumpcja obu konsumentów także będzie stała w czasie i równa pewnym c^1 i $c^2 = 1 - c^1$. Oczekujemy, że $c^1 > c^2$, gdyż konsument pierwszy jest bogatszy – wcześniej otrzymuje dochód. Przyjmując $c_t^1 = c^1$ z warunku pierwszego rzędu uzyskujemy:

$$p_t = \frac{\beta^t u'(c^1)}{\lambda^1}.$$

Po podstawieniu za p_t do ograniczenia budżetowego (pierwszego konsumenta) mamy:

$$\sum_{t=0}^{\infty} \frac{\beta^t u'(c^1)}{\lambda^1} c^1 = \sum_{t=0}^{\infty} \frac{\beta^{2t} u'(c^1)}{\lambda^1}.$$

Stąd:

$$c^1 \sum_{t=0}^{\infty} \beta^t = \sum_{t=0}^{\infty} \beta^{2t},$$

co daje ostatecznie:

$$c^1 = \frac{1}{1 + \beta}.$$

4 Interakcje optymalizujących podmiotów. Równowaga ogólna

Mając c^1 możemy wyznaczyć wektor cen. Oczywiście jeśli $(p_t)_{t=1}^{\infty}$ jest wektorem cen gwarantującym równowagę, to jest nim też $(\alpha p_t)_{t=1}^{\infty}$, dla dowolnego $\alpha > 0$. Możemy więc podzielić cenę $p_t = \beta^t u'(c^1)/\lambda^1$ przez p_0 uzyskując wycenę konsumpcji w okresie t w jednostkach konsumpcji z okresu 0, w tym przypadku $p_t = \beta^t$. Sprawdzenie, że wektor cen $p_t = \beta^t$ i alokacja $c_t^1 = 1/(1 + \beta)$, $c_t^2 = \beta/(1 + \beta)$ tworzą równowagę konkurencyjną pozostawiamy jako ćwiczenie.

Ćwiczenie 4.3 Rozwiąż (oddzielnie) problem każdego z konsumentów z przykładu przyjmując $p_t = \beta^t$. Pokaż, że rozwiązaniem jest rzeczywiście $c_t^1 = 1/(1 + \beta)$ i $c_t^2 = \beta/(1 + \beta)$.

Zauważmy na koniec, że iloraz $p_t/p_{t+1} - 1$ ma interpretację stopy procentowej z okresu t na $t + 1$, bo jeśli zrezygnujemy z jednostki konsumpcji „dzisiaj” będziemy mieć p_t , co pozwoli zakupić p_t/p_{t+1} jednostek konsumpcji „jutro”. W naszym przykładzie $p_t/p_{t+1} - 1 = \beta^{-1} - 1$, czyli stopa procentowa jest równa indywidualnej stopie dyskontującej.⁷ Wynik ten jest konsekwencją stałego dochodu w skali makro. Przypadek niezerowego wzrostu gospodarczego w gospodarce składającej się z dwóch konsumentów pozostawiamy jako zadanie.

4.2.2 Równowaga rekursywna

Rekursywne sformułowanie równowagi polega na rezygnacji z założenia o wymianie w okresie $t = 0$ i zastąpieniu jej wymianą w *każdym z okresów* praw do konsumpcji w okresie następnym. Można powiedzieć, że relacja między oboma sformułowaniami jest identyczna z relacją między metodami rozwiązywania problemów dynamicznej optymalizacji: porównywaniem wariantów w przestrzeni całych wiązek konsumpcji i podejściem sekwencyjnym (przez równania Bellmana).

W przypadku n dóbr ($n > 2$)⁸ problem pojedynczego konsumenta z rozdzielną funkcją użyteczności (tj. postaci $\sum_{i=1}^n u_i(c_i)$) można rozwiązywać konstruując funkcję Lagrange’a i poszukując optymalnego wektora $(c_i)_{i=1}^n$ lub też podejść do problemu sekwencyjnie przyjmując, że konsument najpierw decyduje ile skosztować dobra nr 1, potem dobra nr 2 itd. (patrz ćwiczenie 2.7) Klasyczna mikroekonomiczna analiza równowagi w przypadku n dóbr sprowadza się do jednorazowej wymiany przed przystąpieniem do konsumpcji. Równowaga w problemie dynamicznym z wymianą $t = 0$ jest właśnie zastosowaniem tego klasycznego podejścia. Czas jest w nim potraktowany wyłącznie jako indeks dóbr.

Równowaga rekursywna jest opisem rynku składającego się z podmiotów podejmujących decyzje sekwencyjnie (rozwiązujących swoje równania Bellmana). W mikroekonomicznym ujęciu z n dobrami konsumenci najpierw decydują się ile konsumować dobra nr 1 wymieniając się prawami do dobra nr 2. Jeśli np. j -ty konsument chce skosztować więcej dobra nr 1 niż go ma, sprzedaje prawa do dobra nr 2. Po skosztowaniu dobra nr 1 decyduje ile skosztować dobra nr 2. Jeśli jego konsumpcja dobra nr 2 plus wystawione nań „weksle” przekraczają jego dochód, musi znów pożyczyć (wystawić „weksle” na dobro nr 3); w przeciwnym przypadku sprzedaje swoją nadwyżkę dobra nr 2 i nabywa prawa do dobra nr 3. Zauważmy, że w naszym ujęciu pojawia się możliwość oszustwa zwanego schematem Ponzi’ego. Konsument

⁷ Jeśli przyjąć $\beta = 1/(1 + \theta)$, to $\theta = \beta^{-1} - 1$.

⁸ Gdy $n = 2$ z formalnego punktu widzenia podejścia te właściwie się nie różnią.

4.2 Wymiana w horyzoncie nieskończonym. Równowaga Arrowa-Debreu i rekursywna

może skosztować więcej dobra nr 1 niż wynosi jego dochód wystawiając „weksle” na dobro 2 przekraczające jego możliwości. Zamiast oddawać dobro nr 2 wystawia „weksle” na dobro nr 3 itd. Istotą schematu Ponzi’ego jest finansowanie dzisiejszej konsumpcji (w okresie t) wystawianiem obietnic zwrotu w okresie następnym ($t + 1$), które są realizowane poprzez wystawianie „weksli” na konsumpcję w okresie $t + 2$ i tak „do końca świata”.⁹ Poszukując równowagi rekursywnej będziemy nakładać na konsumentów ograniczenia zwane czasem w literaturze anglojęzycznej NPG (ang. *No Ponzi Game*). W tym momencie wracamy do zagadnienia dynamicznego z nieskończonym horyzontem.

Konsumenci w okresie t wymieniają się prawami do konsumpcji „jutro”. Niech a_t^i oznacza prawa (posiadane lub ze znakiem minus wystawione „weksle”) do konsumpcji w okresie t . a_t jest po prostu wielkością aktywów w danym momencie. Przyjmijmy oznaczenie q_t na cenę nabywanego w okresie t prawa do konsumpcji w okresie następnym ($t + 1$). Międzyokresowe ograniczenie budżetowe konsumenta ma postać:

$$c_t^i + q_t a_{t+1}^i \leq y_t^i + a_t^i. \quad (4.15)$$

Stwierdza ono, że łączna wartość dzisiejszej konsumpcji i zakupionych praw do konsumpcji jutro nie może przewyższać dzisiejszego dochodu i wartości aktywów konsumenta. Nierówność (4.15) nie zabezpiecza przed schematami Ponzi’ego. Konsumpcja dziś może być arbitralnie duża, zaś jutrzejsze aktywa arbitralnie małe. Niech $(p_t)_{t=0}^{\infty}$ będzie wektorem cen Arrowa-Debreu. Wtedy wyrażenie:

$$\Omega_t^i = \frac{\sum_{\tau=t}^{\infty} p_{\tau} y_{\tau}^i}{p_t} \quad (4.16)$$

określa bogactwo (w jednostkach konsumpcji z okresu t) i -tego konsumenta w okresie t . Warunek wykluczający schemat Ponzi’ego możemy zapisać następująco:

$$-a_{t+1}^i \leq \Omega_{t+1}^i. \quad (4.17)$$

Nierówność powyższa stwierdza, że konsument nie może obiecać zwrócić jutro więcej, niż są warte jego całkowite dochody od jutra do nieskończoności. Ograniczenie (4.17) nazywane jest czasem naturalnym limitem długu. Analizę alternatywnej postaci warunku NPG pozostawiamy jako zadanie.

Jak wygląda równanie Bellmana związane z problemem maksymalizacji (4.9)? W naszym przypadku czas jest zmienną stanu (argumentem funkcji wartości), bo określa przyszły strumień dochodów, tak więc funkcję wartości i -tego konsumenta można zapisać w postaci $V_t^i(a_t)$.¹⁰ Zmiennymi decyzyjnymi konsumenta są: dzisiejsza konsumpcja c_t i jutrzejsze aktywa a_{t+1} . Stwierdzamy, iż funkcja wartości określona jest równaniem:

$$V_t^i(a_t) = \max_{c_t^i, a_{t+1}^i} \{u(c_t^i) + \beta V_{t+1}^i(a_{t+1}^i)\}, \quad (4.18)$$

⁹ Przykładem schematu Ponzi’ego jest zaciągnięcie kredytu w pewnym banku na kwotę B , spłnienie go z odsetkami poprzez zaciągnięcie w kolejnym banku kredytu w wysokości $B(1+r)$ i spłata go dzięki kolejnemu kredytowi $B(1+r)^2 \dots$

¹⁰ Modele z produkcją oraz modele stochastyczne można sformułować w sposób omijający czas jako argument funkcji wartości. W czystej wymianie z deterministycznym strumieniem dochodów takie sformułowanie inne niż trywialny przypadek $y_t^i = \text{const}$ nie jest możliwe.

4 Interakcje optymalizujących podmiotów. Równowaga ogólna

przy czym maksymalizacja odbywa się pod warunkami (4.15), (4.17) i $c \geq 0$. Oczywiście jeśli funkcja użyteczności chwilowej spełnia standardowe założenie $u' > 0$, to warunek (4.15) stanie się równością. W takim przypadku możemy przeformułować równanie (4.18) następująco:

$$V_t^i(a_t^i) = \max_{c_t^i} \{u(c_t^i) + \beta V_{t+1}^i((y_t^i + a_t^i - c_t^i)/q_t)\}. \quad (4.19)$$

Przejdźmy do określenia równowagi rekursywnej. Po pierwsze, musimy zagwarantować oczyszczanie się rynku dóbr i rynku kredytowego, a więc spełnienie równości (dla każdego t):

$$\sum_{i=1}^n c_t^i = \sum_{i=1}^n y_t^i \quad (4.20)$$

oraz

$$\sum_{i=1}^n a_t^i = 0. \quad (4.21)$$

Równanie (4.21) stwierdza, że suma zobowiązań ($-\sum_{i:a_t^i < 0} a_t^i$) musi być równa sumie wierzytelności ($\sum_{i:a_t^i > 0} a_t^i$). Zauważmy, że opisując równowagę musi podać początkowy rozkład aktywów $(a_0^i)_{i=1}^n$, gdyż a_t^i jest zmienną stanu dla każdego z konsumentów. Na równowagę rekursywną składają się więc: początkowy rozkład aktywów $(a_0^i)_{i=1}^n$, wektor cen względnych q_t , zestaw funkcji wartości $V_t^i(a_t^i)$ spełniających (4.18) i taki optymalnych polityk $c_t^i(a_t^i)$, $a_{t+1}^i(s_t^i)$, że dla każdego t spełnione są warunki (4.20) i (4.21).

Należy oczekiwać, że przy naturalnym założeniu $\forall_i a_0^i = 0$ (na „początku świata” żaden z konsumentów nie ma ani długów ani wierzytelności) równowaga rekursywna i równowaga Arrowa-Debreu prowadzą do tej samej alokacji. Korzystając z twierdzenia o obwiedni dla (4.19) uzyskujemy:

$$V_t^{ii}(a_t^i) = \beta V_{t+1}^{ii}(a_{t+1}^i)/q_t. \quad (4.22)$$

Z warunku pierwszego rzędu maksymalizacji po prawej stronie (4.19) mamy zaś:

$$u'(c_t^i) = \beta V_{t+1}^{ii}(a_{t+1}^i)/q_t. \quad (4.23)$$

Porównując prawe strony (4.22) i (4.23) otrzymujemy:

$$V_t^{ii}(a_t^i) = u'(c_t^i).$$

Podstawiając w (4.22) $u'(c_t^i)$ i $u'(c_{t+1}^i)$ za $V_t^{ii}(a_t^i)$ i $V_{t+1}^{ii}(a_{t+1}^i)$ odpowiednio, po przekształceniach uzyskujemy równanie Eulera:

$$\frac{\beta u'(c_{t+1}^i)}{u'(c_t^i)} = q_t. \quad (4.24)$$

Jednocześnie przesuwając do przodu (o jeden okres) warunek pierwszego rzędu maksymalizacji w problemie konsumenta mającego możliwość wymiany w okresie $t = 0$ (równanie (4.12)) i dzieląc go obustronnie przez (4.12) otrzymujemy:

$$\frac{\beta^{t+1} u'(c_{t+1}^i)}{\beta^t u'(c_t^i)} = \frac{p_{t+1}}{p_t}.$$

4.3 Twierdzenia ekonomii dobrobytu

Po skróceniu mamy:

$$\frac{\beta u'(c_{t+1}^i)}{u'(c_t^i)} = \frac{p_{t+1}}{p_t}. \quad (4.25)$$

Warunek ten będzie identyczny z (4.24) jeśli tylko będzie zachodziła równość:

$$q_t = \frac{p_{t+1}}{p_t}. \quad (4.26)$$

Niech $(\bar{c}_t^i)_{t=0}^\infty$ będzie ścieżką konsumpcji i -tego konsumenta w równowadze Arrowa-Debreu. Pokażemy, że w równowadze rekursywnej z cenami danymi przez (4.26) konsumenta stać na tę samą ścieżkę. Załóżmy, że konsument nasz ma następującą politykę w stosunku do aktywów:

$$a_{t+1}^i = \frac{\sum_{\tau=t+1}^\infty p_\tau (\bar{c}_\tau^i - y_\tau^i)}{p_{t+1}}, \quad (4.27)$$

a więc oszczędza tyle, ile jego przyszła konsumpcja przekracza przyszłe dochody (wycenione według cen Arrowa-Debreu przeliczonych na okres $t+1$). Z międzyokresowego ograniczenia budżetowego (4.15) na okres $t=0$ mamy:

$$c_0^i + \frac{p_1}{p_0} \frac{\sum_{\tau=1}^\infty p_\tau (\bar{c}_\tau^i - y_\tau^i)}{p_1} = y_0^i + 0.$$

Stąd:

$$p_0 c_0^i = \sum_{\tau=0}^\infty p_\tau y_\tau^i - \sum_{\tau=1}^\infty p_\tau \bar{c}_\tau^i.$$

Konfrontując ten warunek z ograniczeniem budżetowym (4.10) (gdy zachodzi równość) otrzymujemy:

$$c_0^i = \bar{c}_0^i. \quad (4.28)$$

Wychodząc od ograniczenia (4.15) na okres $t=1$ i postępując podobnie uzyskamy $c_1^i = \bar{c}_1^i$ i przez indukcję $\forall_{t \geq 1} c_t^i = \bar{c}_t^i$. Oznacza to, że ścieżka konsumpcji implikowana przez politykę (4.27) pokrywa się ze ścieżką konsumpcji w równowadze Arrowa-Debreu i jest dopuszczalna.

4.3 Twierdzenia ekonomii dobrobytu

Poniżej prezentujemy dwa twierdzenia ekonomii dobrobytu w ogólnej wersji. Nowoczesne ich sformułowanie pochodzi od Arrowa [2], dowód tych twierdzeń w przypadku dowolnych (w tym nieskończeniowymym) przestrzeni liniowych należy do Debreu [9].¹¹ Nasza prezentacja bazuje na artykule Debreu.

¹¹ Artykuły Arrowa i Debreu zostały przedrukowane w [3] i [10].

4 Interakcje optymalizujących podmiotów. Równowaga ogólna

4.3.1 Założenia i definicje

Niech L będzie przestrzenią liniową nad ciałem liczb rzeczywistych \mathbb{R} (patrz definicja C.1). L interpretujemy jako przestrzeń dóbr i czynników produkcji wymienianych w gospodarce. Konsumentów indeksujemy elementami zbioru $\mathcal{I} = \{1, \dots, m\}$ przedsiębiorstwa elementami zbioru $\mathcal{J} = \{1, \dots, n\}$. Każdy konsument dokonuje wyboru swojej wiązki konsumpcji i wiązki wkładów czynników produkcji (ze znakiem minus) x^i ze zbioru $X^i \subset L$. Zbiór X^i ograniczony jest warunkami przeżycia konsumenta oraz warunkami nieujemnego poziomu konsumpcji i niedodatniego poziomu nakładów czynników produkcji.¹² Przedsiębiorstwo wybiera wiązkę wkładów czynników produkcji (ze znakiem minus) i produkowanych dóbr y^j ze zbioru $Y^j \subset L$. Zbiór Y^j w pełni opisuje ograniczenia technologiczne j -tego przedsiębiorstwa, można o nim myśleć jako o geometrycznym odbiciu jego struktury *input-output*.

Definicja 4.6 *Alokacją nazywamy dowolny element $[(x^i), (y^j)] \in X^1 \times \dots \times X^m \times Y^1 \times \dots \times Y^n$, a więc dowolne przypisanie konsumentom i przedsiębiorstwom poziomów konsumpcji, nakładów czynników produkcji i wielkości produkcji.*

Definicja 4.7 *Niech z oznacza początkowy zasób (łącznie wyposażenie początkowe wszystkich uczestników wymiany) w gospodarce. Alokacja $[(x^i), (y^j)]$ jest dopuszczalna, jeśli $\sum_i x^i - \sum_j y^j = z$, a więc popyt netto konsumentów i przedsiębiorstw zaspokajany jest przez egzogeniczny zasób początkowy.*

System cen będziemy utożsamiać z pewnym funkcjonałem liniowym p na L (patrz strona 159).¹³ Interpretacja p jest następująca: px^i określa wartość wiązki i -tego konsumenta, zaś py^j zysk j -tego przedsiębiorstwa.

Celem konsumentów jest osiągnięcie największej satysfakcji z konsumpcji opisanej relacją słabej preferencji \preceq^i na zbiorze X^i , przy zadanym przez p ograniczeniu budżetowym. Niech z^i będzie wyposażeniem początkowym i -tego konsumenta.¹⁴ Będzie on poszukiwał takiego $\bar{x}^i \in X^i$, że

$$p\bar{x}^i = pz^i \quad \wedge \quad \forall_{x^i \in X^i} \bar{x}^i \prec x^i \Rightarrow p\bar{x}^i < px^i. \quad (4.29)$$

Celem przedsiębiorstw jest wybór takich nakładów czynników produkcji i poziomu produkcji by osiągnąć maksymalny zysk. Przedsiębiorstwo poszukuje takiego $\bar{y}^j \in Y^j$, by

$$\forall_{y^j \in Y^j} py^j \leq p\bar{y}^j. \quad (4.30)$$

Definicja 4.8 *Równowagą konkurencyjną nazywamy taki funkcjonal cen p i taką alokację dopuszczalną $[(\bar{x}^i), (\bar{y}^j)]$, przy których spełnione są warunki (4.29) i (4.30).*

¹² Nakłady czynników produkcji brane są ze znakiem minus, a więc ograniczenia na nakłady czynników produkcji oznaczają, że np. nie można pracować mniej niż zero czasu.

¹³ W przypadku skończonej liczby k dóbr L jest przestrzenią \mathbb{R}^k i system cen to wektor k liczb (p_1, \dots, p_k) , którego i -tą składowe interpretujemy jako cenę i -tego dobra. Już w poprzednim podrozdziale prezentowaliśmy model, w którym L nie ma skończonego wymiaru. Właśnie po to, by nasze twierdzenia obejmowały także sytuacje tego typu musimy odwoływać się do wyników analizy funkcjonalnej i określać system cen jako funkcjonal liniowy na L . Oczywiście w przypadku przestrzeni dóbr \mathbb{R}^k funkcjonały liniowe mają reprezentację w postaci wektora (p_1, \dots, p_k) – przestrzeń sprzężona z \mathbb{R}^k jest izomorficzna z \mathbb{R}^k , patrz strony 159-160.

¹⁴ Zakładamy, że cały wyposażenie początkowe z jest w rękach konsumentów, a więc $\sum_i z^i = z$.

4.3 Twierdzenia ekonomii dobrobytu

Poniższa definicja jest naturalnym rozszerzeniem definicji 4.5.

Definicja 4.9 Alokację $[(\bar{x}^i), (\bar{y}^j)]$ nazywamy efektywną w sensie Pareto jeśli jest dopuszczalna i nie istnieje taka alokacja dopuszczalna $[(x^i), (y^j)]$, dla której

$$\forall_{i \in J} \bar{x}^i \preceq^i x^i \quad \wedge \quad \exists_{i \in J} \bar{x}^i \prec^i x^i.$$

W obu twierdzeniach ekonomii dobrobytu musimy wprowadzić kilka dodatkowych założeń o preferencjach konsumentów. Pierwszym z nich będzie założenie, że alokacje w równowadze konkurencyjnej nie pokrywają się z punktami nasycenia żadnego konsumentów.

Definicja 4.10 Wiązkę $x^i \in X^i$ nazywamy punktem nasycenia, jeśli dla każdego $x^{i'} \in X^i$ zachodzi $x^{i'} \preceq^i x^i$.

Bardzo ważnym założeniem jest założenie o wypukłości preferencji.

Definicja 4.11 Mówimy, że preferencje \prec określone na X są wypukłe, jeśli dla każdego $x', x'' \in X$ takich, że $x' \prec x''$ zachodzi:

$$\forall_{0 < t < 1} x' \prec (1-t)x' + tx''. \quad (4.31)$$

Wypukłość preferencji oznacza więc, że jeśli wiązka x'' jest lepsza od wiązki x' , to wszystkie wiązki pośrednie (na „odcinku” łączącym x' z x'') są także lepsze niż x' . Jak ma się założenie o wypukłości preferencji do standardowego założenia o wklęsłości funkcji użyteczności? Rozpatrzmy przykład z jednym dobrem w dwóch okresach i dyskontowaniem użyteczności. Niech więc $L = \mathbb{R}^2$ i $X = \mathbb{R}_+^2$. Preferencje na X zadane są przez następującą funkcję:

$$U(x) = u(x_0) + \beta u(x_1),$$

gdzie u jest wklęsłą funkcją użyteczności chwilowej. Przyjmijmy, że $U(x) < U(x')$, czyli $x \prec x'$. Pokażemy, że dla dowolnego $t \in (0, 1)$ zachodzi $U(x) < U(tx + (1-t)x')$ (czyli $x \prec tx + (1-t)x'$). Mamy bowiem:

$$\begin{aligned} U(tx + (1-t)x') &= u(tx_0 + (1-t)x'_0) + \beta u(tx_1 + (1-t)x'_1) > \\ &tu(x_0) + (1-t)u(x'_0) + \beta tu(x_1) + (1-t)\beta u(x'_1) = \\ &tU(x) + (1-t)U(x') > U(x). \end{aligned}$$

Pierwsza nierówność wynika z wklęsłości u , druga z założenia $U(x) < U(x')$.¹⁵

Ćwiczenie 4.4 Niech L będzie przestrzenią ciągów ograniczonych l^∞ i niech $X = \{a_t \in l^\infty : \forall_t a_t > 0\}$. Pokaż, że preferencje zadane na X przez:

$$\sum_{t=0}^{\infty} \beta^t u(c_t),$$

gdzie u jest wklęsłą funkcją użyteczności chwilowej, są wypukłe.

¹⁵ Niech $a < b$ i $t \in (0, 1)$. Mamy wtedy: $ta + (1-t)b = ta + (1-t)b - (1-t)a + (1-t)a = a + (1-t)(b-a) > a$.

4 Interakcje optymalizujących podmiotów. Równowaga ogólna

Wprowadzimy jeszcze jeden warunek regularności preferencji.

Jeśli zbiór $X \subset L$ jest zbiorem wypukłym, wtedy dla każdego dwóch punktów $x', x'' \in X$ zbiór $\{t \in \mathbb{R}: tx' + (1-t)x'' \in X\}$ jest przedziałem w \mathbb{R} (być może bez jednego lub dwóch punktów końcowych).¹⁶

Definicja 4.12 Niech \preceq będą pewnymi preferencjami określonymi na X . Mówimy, że preferencje \preceq spełniają słaby warunek ciągłości, jeśli dla dowolnych trzech punktów $x, x', x'' \in X$ zbiory

$$\{t: (1-t)x' + tx'' \in X \wedge x \preceq (1-t)x' + tx''\}$$

i

$$\{t: (1-t)x' + tx'' \in X \wedge (1-t)x' + tx'' \preceq x\}$$

są domknięte w:

$$\{t: (1-t)x' + tx'' \in X\}.$$

Lemat 4.1 Niech X będzie wypukłym podzbiorem pewnej przestrzeni liniowej L , \prec będą określonymi na X wypukłymi preferencjami spełniającymi słaby warunek ciągłości. Wtedy dla każdego $x', x'' \in X$ takich, że $x' \preceq x''$ zachodzi:

$$\forall_{0 \leq t \leq 1} x' \preceq (1-t)x' + tx''. \quad (4.32)$$

Dowód. Ze słabej ciągłości preferencji wynika, że zbiór

$$\{t: (1-t)x' + tx'' \in X \wedge (1-t)x' + tx'' \prec x'\}$$

jest otwarty w

$$\{t: (1-t)x' + tx'' \in X\}$$

i otwarte jest jego przecięcie z przedziałem $(0, 1)$. Pokażemy, że ta część wspólna jest pusta. W przeciwnym przypadku należałyby do niej dwie liczby $t_1 < t_2$. Odpowiadające tym liczbom punkty x^1, x^2 musiałyby spełniać $x^1 \prec x' \preceq x''$ oraz $x^2 \prec x' \preceq x''$. Wypukłości preferencji w połączeniu z warunkiem $x^1 \prec x''$ implikuje $x^1 \prec x^2$ (x^2 jest pomiędzy x^1 i x''). Analogicznie, z $x^2 \prec x'$ wynika $x^2 \prec x^1$. Uzyskana sprzeczność dowodzi tezy. ■

Dowód poniższego faktu pozostawiamy jako ćwiczenie.

Fakt 4.1 Niech X będzie wypukłym podzbiorem pewnej przestrzeni liniowej L , \prec będą określonymi na X wypukłymi preferencjami spełniającymi słaby warunek ciągłości i niech x' będzie dowolną wiązką z X . Zbiory:

$$\overline{X}(x') \equiv \{x \in X: x' \preceq x\}$$

$$\overline{\overline{X}}(x') \equiv \{x \in X: x' \prec x\}$$

są wypukłe ($\overline{\overline{X}}(x')$ może być pusty).

Ćwiczenie 4.5 Korzystając z lematu 4.1 udowodnij powyższy fakt.

¹⁶ Zbiór $\{t \in \mathbb{R}: tx' + (1-t)x'' \in X\}$ jest zbiorem tych parametrów t , dla których prosta w L przechodząca przez punkty x' i x'' przecina się z X . Ponieważ X jest zbiorem wypukłym żadna prosta nie może z niego „wyjść” i ponownie „wejść”, a więc zbiór parametrów dla których $tx' + (1-t)x'' \in X$ może być tylko odcinkiem, półprostą lub prostą.

4.3.2 Pierwsze twierdzenie ekonomii dobrobytu

Twierdzenie 4.1 Niech będą spełnione następujące warunki:

1. dla każdego i zbiór X^i jest wypukły,
2. dla każdego i preferencje \prec^i są wypukłe.

Wtedy każda taka alokacja w równowadze konkurencyjnej $[(\bar{x}^i), (\bar{y}^j)]$, w której żadna z wiązek \bar{x}^i nie jest punktem nasycenia w X^i jest efektywna w sensie Pareto.

Dowód. Z warunku optymalności wyboru konsumenta przy zadanym ograniczeniu budżetowym mamy ($\bar{x}^i, x^i \in X^i$):

$$\bar{x}^i \prec^i x^i \Rightarrow px^i > p\bar{x}^i.$$

Niech wiązka x^i spełnia warunek:

$$\bar{x}^i \simeq^i x^i \iff (\bar{x}^i \preceq^i x^i) \wedge (x^i \preceq^i \bar{x}^i),$$

czyli będzie równoważna z \bar{x}^i .¹⁷ Ponieważ \bar{x}^i nie jest punktem nasycenia w X^i istnieje $x^{i'} \in X^i$ preferowane nad \bar{x}^i , a więc także nad x^i . Z wypukłości preferencji, dla $0 < t < 1$ mamy:

$$\bar{x}^i \prec^i (1-t)x^i + tx^{i'},$$

co w połączeniu z założeniem optymalnego wyboru przy ograniczeniu budżetowym daje:

$$p\bar{x}^i < (1-t)px^i + tpx^{i'}.$$

Przechodząc do granicy przy $t \rightarrow 0$ wnioskujemy, iż:

$$(\bar{x}^i \simeq^i x^i) \Rightarrow px^i \geq p\bar{x}^i.$$

Celem naszym jest pokazanie, że żadna alokacja $[(x^i), (y^j)]$, dla której

$$\forall_{i \in J} \bar{x}^i \preceq^i x^i \quad \wedge \quad \exists_{i \in J} \bar{x}^i \prec^i x^i,$$

nie jest dopuszczalna. Niech i' będzie takim indeksem, że $\bar{x}^{i'} \prec^{i'} x^{i'}$. Z założenia o maksymalizacji użyteczności oraz poprzednich rozważań wynika, że $px^{i'} > p\bar{x}^{i'}$, a dla $i \neq i'$ zachodzi $px^i \geq p\bar{x}^i$. Dodając stronami te nierówności uzyskujemy:

$$\sum_i px^i > \sum_i p\bar{x}^i.$$

Z warunku maksymalizacji zysków przedsiębiorstwa mamy:

$$\sum_j py^j \leq \sum_j p\bar{y}^j.$$

Powyższe nierówności łącznie dają:

$$\sum_i px^i - \sum_j py^j > \sum_i p\bar{x}^i - \sum_j p\bar{y}^j.$$

¹⁷ Istnieje przynajmniej jedna taka wiązka, gdyż na mocy zwrotności relacji słabej preferencji \bar{x}^i jest równoważna ze sobą.

4 Interakcje optymalizujących podmiotów. Równowaga ogólna

Z liniowości funkcjonału p i dopuszczalności alokacji $[(\bar{x}^i), (\bar{y}^j)]$ wynika:

$$\sum_i p\bar{x}^i - \sum_j p\bar{y}^j = p \left(\sum_i \bar{x}^i - \sum_j \bar{y}^j \right) = pz.$$

Ponieważ:

$$p \left(\sum_i x^i - \sum_j y^j \right) > pz,$$

$\sum_i x^i - \sum_j y^j \neq z$ i wnioskujemy, iż alokacja $[(x^i), (y^j)]$ nie jest dopuszczalna. ■

4.3.3 Drugie twierdzenie ekonomii dobrobytu

Twierdzenie 4.2 Niech L będzie przestrzenią Banacha (ogólniej: przestrzenią liniowo topologiczną). Załóżmy, że spełnione są warunki 1 i 2 twierdzenia 4.1 oraz:

3. preferencje spełniają słaby warunek ciągłości,
4. zbiór $Y = \sum_i Y_i \equiv \{y \in L : y = \sum_i y^i, y^i \in Y^i\}$ jest wypukły,

5. L jest skończonego wymiaru lub Y ma niepuste wnętrze.

Wtedy każdej alokacji $[(\bar{x}^i), (\bar{y}^j)]$ efektywnej w sensie Pareto, w której dla pewnego i wiązka \bar{x}^i nie jest punktem nasycenia, odpowiada (nietrywialny, tj. nierówny tożsamościowo zeru) funkcjonał liniowy p na L taki, że:

$$\forall_i (x^i \in X^i) \wedge (\bar{x}^i \preceq x^i) \Rightarrow px^i \geq p\bar{x}^i \quad (4.33)$$

oraz

$$\forall_i y^i \in Y^i \Rightarrow py^i \leq p\bar{y}^i. \quad (4.34)$$

Dowód. Z faktu 4.1 wynika, że zbiory:

$$\overline{X^i}(\bar{x}^i) = \{x^i \in X^i : \bar{x}^i \preceq x^i\}$$

i

$$\overline{\overline{X^i}}(\bar{x}^i) = \{x^i \in X^i : \bar{x}^i \prec x^i\}$$

są wypukłe.

Niech i' będzie takim indeksem, dla którego wiązka $\bar{x}^{i'}$ nie jest punktem nasycenia.

Wtedy $\overline{\overline{X^{i'}}}(\bar{x}^{i'})$ jest niepusty. Określmy zbiór Z następująco:

$$Z = \overline{\overline{X^{i'}}}(\bar{x}^{i'}) + \sum_{i \neq i'} \overline{X^i}(\bar{x}^i) - \sum_j Y^j.$$

Z wypukłości składników sumy wynika, iż Z jest zbiorem wypukłym i jeśli Y ma niepuste wnętrze ma je też i Z . Ponieważ wyjściowa alokacja $[(\bar{x}^i), (\bar{y}^j)]$ była efektywna w sensie Pareto, zasób początkowy z nie może należeć do Z . Twierdzenie o oddzielaniu (twierdzenie C.15) zapewnia istnienie takiego funkcjonału liniowego p , że $pz \leq pz' : z' \in Z$. Ponieważ:

$$z = \sum_i \bar{x}^i - \sum_j \bar{y}^j$$

4.3 Twierdzenia ekonomii dobrobytu

dla każdych $x^{i'} \in \overline{X^{i'}}$, $x^i \in \overline{X^i}$ ($i \neq i'$), więc:

$$p \left[\sum_i (x^i - \bar{x}^i) - \sum_j (y^j - \bar{y}^j) \right] \geq 0. \quad (4.35)$$

Rozpatrzmy punkty $(1-t)\bar{x}^i + tx^{i'} \in \overline{X^{i'}}$, gdzie $0 < t < 1$. Przechodząc do granicy przy $t \rightarrow 0$ stwierdzamy, że powyższa nierówność zachodzi także, dla $x^{i'} \in \overline{X^{i'}}$. Weźmy teraz dowolne i' i ustalmy $x^i = \bar{x}^i$: $i \neq i'$, $x^{i'} \in \overline{X^{i'}}$ oraz $\forall_j y^j = \bar{y}^j$. Z (4.35) wynika wtedy

$$px^{i'} \geq p\bar{x}^{i'}.$$

Ponieważ indeks i' był wybrany dowolnie, więc wykazaliśmy (4.33). By wykazać (4.34) wystarczy przyjąć $\forall_i x^i = \bar{x}^i$ i ustalić wszystkie wiązki y^j poza jedną na poziomie $y^j = \bar{y}^j$. ■

Zwróćmy uwagę, iż twierdzenie 4.2 jest tylko częściowym odwróceniem twierdzenia 4.1. Maksymalizacja użyteczności oznacza, iż wszystkie lepsze wiązki konsumpcji są droższe (konsumenta nie stać na nie). W tezie drugiego twierdzenia ekonomii dobrobytu mamy zaś stwierdzenie, iż wiązki niegorsze niż \bar{x}^i nie są tańsze. Pełne odwrócenie tezy pierwszego twierdzenia dobrobytu jest możliwe przy pewnych dodatkowych założeniach i zawarte jest w poniższym wniosku.

Wniosek 4.1 Niech $[(\bar{x}^i), (\bar{y}^j)]$ będzie alokacją efektywną w sensie Pareto i p będzie funkcjonalem, o którym mowa w tezie twierdzenia 4.2. Niech dodatkowo dla każdego i istnieje $x^{i'} \in X^i$, takie, że $p\bar{x}^i > px^{i'}$. Wtedy alokację $[(\bar{x}^i), (\bar{y}^j)]$ można uzyskać jako wynik równowagi konkurencyjnej, przy takiej redystrybucji zasobów początkowych, by dla każdego i zachodziła nierówność $pz^i \geq p\bar{x}^i$.

Dowód. Rozpatrzmy $x^i \in X^i$, takie, że $px^i \leq p\bar{x}^i$. Niech $0 < t < 1$ wtedy

$$p[(1-t)x^i + tx^{i'}] < p\bar{x}^i.$$

Stąd i z (4.33) wynika $(1-t)x^i + tx^{i'} \prec^i \bar{x}^i$. Zbiór

$$\{t: (1-t)x^i + tx^{i'} \in X^i \wedge (1-t)x^i + tx^{i'} \preceq \bar{x}^i\}$$

zawiera przedział otwarty $(0, 1)$ i jest domknięty w $\{t: (1-t)x^i + tx^{i'} \in X^i\}$, a więc zawiera 0. Stąd $x^i \preceq \bar{x}^i$. Ponieważ wiązka x^i była wybrana dowolnie, pokazaliśmy, że każdy niedroższy koszyk jest nielepszy, co znaczy, że \bar{x}^i jest optymalnym wyborem przy ograniczeniu budżetowym zadanym przez p . Nierówność (4.34) jest warunkiem maksymalizacji zysków przedsiębiorstw, a więc system cen p i alokacja $[(\bar{x}^i), (\bar{y}^j)]$ spełniają warunki definicji równowagi konkurencyjnej. ■

4.3.4 Uwagi

Nasze rozważania bazowały na pojęciu preferencji, a nie samej funkcji użyteczności choć jak wskazywaliśmy na stronie 53 użyteczność w naturalny sposób zadaje relację preferencji. Czytelnik nie powinien jednak ulec wrażeniu, że operowanie

4 Interakcje optymalizujących podmiotów. Równowaga ogólna

na preferencjach istotnie zwiększa ogólność naszych rozważań, gdyż przyjęliśmy całkiem mocne założenia o preferencjach (wypukłość i słaby aksjomat ciągłości).

W klasycznym artykule Debreu [9] jak i w większości podręcznikowych prezentacji w definicji równowagi konkurencyjnej nie mówi się nic o rozkładzie początkowego zasobu. Odwoływanie się do rozkładu (z^i) w definicji równowagi nie ma znaczenia dla dowodu twierdzeń 4.1 i 4.2. Rozkład zasobu początkowego pojawia się tylko we wniosku 4.1. Wydaje się, że takie odejście od klasycznej prezentacji pozwala jednak lepiej zrozumieć konsekwencje twierdzeń ekonomii dobrobytu, szczególnie drugiego twierdzenia i płynącego zeń wniosku.

Geometryczną interpretację pierwszego twierdzenia dobrobytu już podawaliśmy omawiając rysunek 4.3b. Zauważmy, że idea stojąca za tym twierdzeniem jest całkiem prosta: poprawa sytuacji któregoś z konsumentów łamiąc jego ograniczenie budżetowe musiałaby jednocześnie łamać ograniczenie zasobów.

Przyjrzyjmy się jeszcze przez chwilę rysunkowi 4.2b. Drugie twierdzenie ekonomii dobrobytu mówi, że przez każdy punkt \bar{x}^i współtworzący alokację efektywną w sensie Pareto można przeprowadzić prostą (w ogólnym przypadku hiperpłaszczyzną $px^i = pz^i$), tak, że dla każdego konsumenta wszystkie wiązki dóbr preferowane nad wyjściowy będą leżały „po jednej stronie” rzeczonej prostej (hiperpłaszczyzny). Jednocześnie rysunek 4.2b pozwala zrozumieć kwestię początkowych zasobów i ewentualnie redystrybucji, o której mowa we wniosku 4.1. Jak już zauważyliśmy wcześniej przy początkowym zasobie jak na rysunku 4.2b alokacja P_2 nie może zostać osiągnięta w wyniku wymiany. By osiągnąć alokację P_2 należy najpierw dokonać transferu od pierwszego konsumenta do drugiego. Oczywiście sama kwestia wielkości transferu jest trywialna – wystarczy tak rozdzielić zasób początkowy by odpowiadał on oczekiwanej przez nas alokacji Pareto-efektywnej. Wtedy równowaga konkurencyjna ustali się w tym samym punkcie (w przypadku czystej wymiany do handlu *de facto* nie dojdzie). Zauważmy, że drugie twierdzenie ekonomii dobrobytu zupełnie oddziela kwestię efektywności od problemów sprawiedliwości społecznej i skali redystrybucji.

Drugie twierdzenie ekonomii dobrobytu oprócz znaczenia dla „ekonomii politycznej” odgrywa także dużą rolę praktyczną. Załóżmy, że interesujący nas model gospodarki spełnia odpowiednie założenia i wnioskujemy, iż każdą alokację efektywną można uzyskać jako wynik równowagi konkurencyjnej. Jeśli umiemy relatywnie prosto określać efektywne alokacje, np. jako rozwiązania problemu „centralnego planisty”, można pominąć o wiele trudniejsze (także numerycznie) szukanie równowagi konkurencyjnej, wiedząc, że (przy odpowiedniej alokacji zasobów początkowych) osiągnęlibyśmy ten sam wynik.¹⁸ Powyższe podejście było standardowo wykorzystywane m. in. przez teoretyków realnego cyklu koniunkturalnego, aż do wprowadzenia do modeli RBC elementów łamiących założenia twierdzenia ekonomii dobrobytu, lub, inaczej mówiąc, do czasów budowy modeli, w których równowaga konkurencyjna nie jest efektywna w sensie Pareto.

¹⁸ Ten tok rozumowania prowadzi do ciekawego wniosku. W niniejszym rozdziale jak i w całej książce zakładamy istnienie równowagi konkurencyjnej. Jednak w sytuacji, w której umiemy znaleźć alokacje Pareto-efektywne i spełnione są odpowiednie założenia, drugie twierdzenie ekonomii dobrobytu i płynący zeń wniosek dowodzą jej istnienia.

4.4 Centralny planista i koncepcja reprezentatywnego podmiotu

Rozpatrzmy jeszcze raz przykład czystej wymiany z nieskończonym horyzontem ze strony 57. Gospodarka składa się z n konsumentów żyjących nieskończenie długo. Użyteczność i -tego konsumenta, jaką zapewnia mu strumień konsumpcji $(c_t^i)_{t=0}^\infty$ jedynego dobra wynosi:

$$U^i = \sum_{t=0}^{\infty} \beta^t u(c_t^i), \quad (4.36)$$

Konsument ten ma zapewniony strumień dochodów $(y_t^i)_{t=0}^\infty$.

Załóżmy, że alokacji dokonuje instytucja o charakterze nakazowo-rozdzielczym – „centralny planista”, który rozwiązuje następujący problem (tzw. problem Pareto):

$$\max U = \sum_{i=1}^n \theta^i U^i \quad \text{pw.} \quad \forall t \quad \sum_{i=1}^n c_t^i \leq \sum_{i=1}^n y_t^i \equiv y_t. \quad (4.37)$$

O wagach θ^i zakładamy, że są dodatnie i $\sum_{i=1}^n \theta^i = 1$. Centralny planista bierze pod uwagę całkowite zasoby w gospodarce (a więc zbiór rozwiązań dopuszczalnych jego problemu stanowią alokacje dopuszczalne) i rozdziela je tak, by maksymalizować średnią ważoną użyteczność całego społeczeństwa.

Łatwo zauważyć, że alokacja dokonana przez centralnego planistę będzie zawsze efektywna w sensie Pareto. Niech $[(\bar{c}_t^i)_{t=0}^\infty]_{i=1}^n$ będzie rozwiązaniem problemu (4.37). Jeśli alokacja ta nie byłaby efektywna w sensie Pareto, wtedy istniałaby taka alokacja dopuszczalna, $[(\bar{c}_t^i)_{t=0}^\infty]_{i=1}^n$, dla której $\forall_i U^i((\bar{c}_t^i)_{t=0}^\infty) \geq U^i((\bar{c}_t^i)_{t=0}^\infty)$ i dla pewnego i byłoby $U^i((\bar{c}_t^i)_{t=0}^\infty) > U^i((\bar{c}_t^i)_{t=0}^\infty)$. Oznaczałoby to, że

$$U \left([(\bar{c}_t^i)_{t=0}^\infty]_{i=1}^n \right) > U \left([(\bar{c}_t^i)_{t=0}^\infty]_{i=1}^n \right),$$

co przeczy założeniu o $[(\bar{c}_t^i)_{t=0}^\infty]_{i=1}^n$.

Na mocy drugiego twierdzenia ekonomii dobrobytu i płynącego zeń wniosku alokację $[(\bar{c}_t^i)_{t=0}^\infty]_{i=1}^n$ można uzyskać jako wynik równowagi konkurencyjnej. W szczególności, jeśli konsumenci mają identyczne preferencje i wyposażenia początkowe (strumienie dochodów) rozwiązanie problemu centralnego planisty (dla $\theta^i \equiv 1/n$) będzie się pokrywało z równowagą konkurencyjną, która w tym szczególnym przypadku nosi nazwę symetrycznej.¹⁹

Niech $u^i(\cdot)$ będą niekoniecznie identycznymi funkcjami użyteczności chwilowej konsumentów. Określmy następującą funkcję:

$$u(c) = \max \left\{ \sum_{i=1}^n \theta^i u^i(c^i) \quad \text{pw.} \quad c^1 + \dots + c^n \leq c \right\}. \quad (4.38)$$

Pokażemy, że $u(\cdot)$ jest funkcją rosnącą i wklęsłą. Lagranżan problemu optymalizacyjnego (4.38) ma postać:

$$\mathcal{L} = \max \sum_{i=1}^n \theta^i u^i(c^i) - \lambda \left(\sum_{i=1}^n c^i - c \right).$$

¹⁹ Oczywiście, w czystej wymianie rozwiązanie problemu centralnego planisty z równymi wagami, jak i określenie równowagi symetrycznej są trywialne.

4 Interakcje optymalizujących podmiotów. Równowaga ogólna

Warunkiem pierwszego rzędu, ze względu na c^i jest układ równań:

$$\theta^i u^{ii}(c^i) = \lambda.$$

Niech liczby $c^{*1}, \dots, c^{*n}, \lambda^*$ będą optymalne. Z definicji $u(c)$ i twierdzenia o obwiedni (patrz też strona 24) otrzymujemy:

$$u'(c) = \frac{\partial}{\partial c} \mathcal{L}(c^{*1}, \dots, c^{*n}, \lambda^*) = \lambda^*.$$

Mamy $\lambda^* = \theta^i u^{ii}(c^{*i})$. Z $u^{ii} > 0$ wynika wprost, że $u' > 0$. Wiemy też, że u^{ii} jest funkcją malejącą. Jeśli w problemie (4.38) wzrośnie c , wzrośnie też c^i , więc zmaleje λ^* , z czego wnioskujemy o wklęsłości $u(\cdot)$.

Korzystając z funkcji u możemy (odwołując się do odpowiednich własności maksimum) przeformułować problem centralnego planisty następująco:

$$\max \sum_{i=1}^n u(c_i) \quad \text{pw.} \quad \forall_t c_t \leq y_t. \quad (4.39)$$

Można powiedzieć, że centralny planista maksymalizuje użyteczność *reprezentatywnego podmiotu*.

Zadania

Zadanie 4.1 Na rynku spotyka się dwóch konsumentów o tym samym współczynniku dyskontującym $0 < \beta < 1$ i chwilowej funkcji użyteczności $u(c) = \ln c$ dysponujących zasobami $(y_t^1)_{t=0}^\infty = (1, 0, G^2, 0, G^4, 0, G^6, \dots)$ i $(y_t^2)_{t=0}^\infty = (0, G, 0, G^3, 0, G^5, 0, \dots)$, przy czym $G > 1$. Podaj definicję równowagi konkurencyjnej z wymianą tylko w okresie $t = 0$. Zapisz warunki jakie ją określają. Zgadnij, że konsumpcja obu konsumentów będzie rosła w stałym tempie G i znajdź równowagę. Pokaż, że jeśli nadać wyrażeniu $p_t/p_{t+1} - 1$ interpretację stopy procentowej r , przyjmą $\theta = \beta^{-1} - 1$ (indywidualna stopa dyskontująca) i $g = G - 1$ (stopa wzrostu), to zachodzi $r \approx \theta + g$. Jaki wpływ na ten wynik ma przyjęta formuła funkcji użyteczności, w szczególności, co by się stało gdybyśmy przyjęli $u(c) = c^{1-\sigma}/(1-\sigma)$ ($\sigma > 0 \wedge \sigma \neq 1$)?

Zadanie 4.2 Warunek NPG często bywa zapisywany następującą nierównością:

$$\lim_{t \rightarrow \infty} \left(\prod_{\tau=0}^{t-1} q_\tau \right) a_t \geq 0, \quad (4.40)$$

którą interpretuje się następująco: zdyskontowane na dziś aktywa na „końcu świata” (w nieskończoności) muszą być nieujemne. Przyjmij, że zachodzi (4.26). Pokaż, że:

$$a_t = \frac{\sum_{\tau=0}^{t-1} p_\tau (y_\tau - c_\tau)}{p_t},$$

jest rozwiązaniem równania

$$c_t^i + q_t a_{t+1}^i = y_t^i + a_t^i$$

z warunkiem początkowym $a_0 = 0$, jeśli tylko $q_t = \frac{p_{t+1}}{p_t}$. Pokaż, że przy tak określonym a_t spełnienie (4.17) i (4.40) wynika wprost z ograniczenia budżetowego (4.10).

Rozdział 5

Konsumpcja i inwestycje w równowadze ogólnej

Celem niniejszego rozdziału jest analiza kilku wariantów zdecentralizowanej wersji modelu Ramseya.¹ Rozdział ten stanowi wprowadzenie do omawianych później stochastycznych modeli równowagi ogólnej (w szczególności teorii realnego cyklu koniunkturalnego) i nowoczesnej teorii wzrostu.

Podstawowy nacisk położony jest na zrozumienie roli poszczególnych założeń, pewne aspekty „techniczne” i interpretację uzyskanych wyników. Czytelnika zainteresowanego implikacjami ekonomicznymi czy też sposobami wprowadzenia endogenicznego postępu technologicznego odsyłamy do literatury wzrostu.²

5.1 Podstawowe założenia

W gospodarce działa wiele podmiotów funkcjonujących nieskończenie długo czerpiących użyteczność z konsumpcji jednego dobra. Zamiast określać je mianem konsumentów i zakładać, że ludzie są nieśmiertelni, będziemy utożsamiać owe podmioty z gospodarstwami domowymi (rodzinami), których liczebność jest stała w czasie. Zakładamy, że żyjący członkowie rodziny cenią konsumpcję przyszłych pokoleń tak, jak własną – odpowiednio odroczoną w czasie. Użyteczność gospodarstwa domowego jest funkcją strumienia konsumpcji reprezentatywnego członka rodziny i wyraża się wzorem:

$$\sum_{t=0}^{\infty} \beta^t u(c_t^i). \quad (5.1)$$

W każdym okresie każde gospodarstwo domowe sprzedaje na rynku jednostkę pracy.

¹ Ścisłej: modelu Ramseya-Cassa-Koopmansa, patrz przypis na stronie 30.

² Patrz np. Barro i Sala-i-Martin [5], Aghion i Howitt [1]. Czytelnikowi należy się jedna uwaga: wielu teoretyków wzrostu posługuje się modelami w czasie ciągłym, co wymaga znajomości rachunku wariacyjnego i/lub zasady maksimum (hamiltoniany). Nieco skróciwą prezentację podstawowych modeli wzrostu w czasie dyskretnym można znaleźć u Sargenta i Ljungqvista [21], rozdz. 11.

5 Konsumpcja i inwestycje w równowadze ogólnej

Przedsiębiorstwa dysponują technologią opisaną ciągłą i dwukrotnie różniczkowalną funkcją $F(K, L)$, gdzie K to nakład kapitału, L – nakład pracy. Funkcja produkcji cechuje się stałymi korzyściami skali (jest jednorodna stopnia 1):³

$$F(\lambda K, \lambda L) = \lambda F(K, L).$$

Z twierdzenia Eulera o funkcjach jednorodnych⁴ mamy:

$$F(K, L) = F_K(K, L)K + F_L(K, L)L. \quad (5.2)$$

Tak więc produkt przy technologii o stałych korzyściach skali można podzielić na części „otrzymane” dzięki nakładom pracy i kapitału. Wyrażenia $MPL \equiv F_L(K, L)$ i $MPK \equiv F_K(K, L)$ interpretujemy jako krańcowy produkt pracy i kapitału.

Naturalne jest założenie, iż krańcowe produkty obu czynników produkcji są dodatnie i maleją wraz z ich nakładami:

$$MPL \equiv F_L, MPK \equiv F_K > 0, \quad F_{KK} = MPK_K, F_{LL} = MPL_L < 0.$$

Różniczkując równość (5.2) po K mamy także:

$$F_K = F_{KK}K + F_K + F_{LK}L + F_LL.$$

Z powyższego równania i z założenia $F_{KK} < 0$ wynika, że pochodne mieszane funkcji produkcji (na mocy twierdzenia Schwarz'a $F_{LK} = F_{KL}$) są dodatnie, co ma prostą interpretację ekonomiczną: wzrost nakładów jednego z czynników produkcji zwiększa krańcowy produkt drugiego (czynniki produkcji są komplementarne).

Często przyjmuje się także dodatkowe założenia o funkcji produkcji:

$$\lim_{K \rightarrow 0} F_K(K, 1) = +\infty, \quad \lim_{L \rightarrow 0} F_L(1, L) = +\infty,$$

$$\lim_{K \rightarrow +\infty} F_K(K, 1) = 0, \quad \lim_{L \rightarrow +\infty} F_L(1, L) = 0.$$

Warunki te (tzw. warunki Inady) mają charakter techniczny, ostatecznie dwa z nich należy odczytywać następująco: przy ustalonym poziomie jednego z czynników produkcji i bardzo dużych nakładach drugiego, krańcowy jego przychód jest znikomy.

³ Ogólnie: mówimy, że funkcja $z(x_1, \dots, x_n)$ jest dodatnio jednorodna stopnia α , jeśli dla każdego dodatnich x_1, \dots, x_n i λ spełniony jest warunek $z(\lambda x_1, \dots, \lambda x_n) = \lambda^\alpha z(x_1, \dots, x_n)$.

⁴ Niech $z(x_1, \dots, x_n)$ będzie funkcją dodatnio jednorodną stopnia α . Ustalmy x_1, \dots, x_n . Dla dowolnego $t > 0$ mamy:

$$z(tx_1, \dots, tx_n) = t^\alpha z(x_1, \dots, x_n).$$

Różniczkując powyższą równość po t uzyskujemy:

$$z_1(tx_1, \dots, tx_n)x_1 + \dots + z_n(tx_1, \dots, tx_n)x_n = \alpha t^{\alpha-1} z(x_1, \dots, x_n).$$

W szczególności dla $t = 1$ mamy:

$$z_1(x_1, \dots, x_n)x_1 + \dots + z_n(x_1, \dots, x_n)x_n = \alpha z(x_1, \dots, x_n).$$

Powyższa równość zachodzi dla dowolnych x_1, \dots, x_n .

5.2 Podstawowa wersja modelu

W wielu analizach korzysta się z tzw. intensywnej postaci funkcji produkcji. Jest ona określona następująco (przyjmujemy $k \equiv \frac{K}{L}$):

$$f(k) \equiv F\left(\frac{K}{L}, 1\right) = \frac{F(K, L)}{L}. \quad (5.3)$$

Funkcja produkcji w postaci intensywnej wyznacza wielkość produkcji na jednostkę pracy w zależności od przypadającego na nią zasobu kapitału. Prostem rachunkiem można sprawdzić następujące równości:

$$F_K(K, L) = f'(k), \quad F_L(K, L) = f(k) - kf'(k). \quad (5.4)$$

Ćwiczenie 5.1 Udowodnij powyższe wzory.

Ćwiczenie 5.2 Pokaż, że funkcja Cobba-Douglasa określona równaniem:

$$F(K, L) = CK^\alpha L^{1-\alpha}.$$

spełnia wszystkie warunki opisujące funkcję produkcji. Jaka jest elastyczność produkcji względem nakładów kapitału i pracy?

5.2 Podstawowa wersja modelu

W najprostszym ujęciu zakładamy, że konsumenci są właścicielami kapitału i wraz z pracą wynajmują go działającą w danym okresie przedsiębiorstwom.⁵ Kapitał w posiadaniu i -tego gospodarstwa domowego ewoluuje zgodnie ze wzorem:

$$k_{t+1}^i = (1 - \delta)k_t^i + i_t^i, \quad (5.5)$$

gdzie δ to stopa deprecjacji, zaś i_t^i wielkość poczynionych przez gospodarstwo inwestycji (brutto). Dobro inwestycyjne jest tym samym co dobro konsumpcyjne (można też przyjąć, iż dobro konsumpcyjne zamienia się na inwestycyjne w proporcji 1:1).

Rynki dobra finalnego i czynników produkcji są konkurencyjne. Działające w t -tym okresie j -te przedsiębiorstwo maksymalizuje zysk:

$$\max \Pi_t^j = F(k_t^j, l_t^j) - w_t l_t^j - r_t k_t^j, \quad (5.6)$$

gdzie w_t określa wynagrodzenie pracy, r_t - wynagrodzenie kapitału, zaś k_t^j i l_t^j to decyzje przedsiębiorstwa o wielkościach wynajętego kapitału i pracy odpowiednio. Zarówno w_t , jak i r_t wyrażone są w jednostkach konsumpcji w okresie t . Jest jasne, że w równowadze zyski przedsiębiorstw będą równe zero.

⁵ Można myśleć, że przedsiębiorstwa działają tylko przez jeden okres. Alternatywnie można założyć, że działają one nieskończenie długo i maksymalizują zdyskontowany w jakiś sposób strumień zysków. Ponieważ jednak w każdym okresie decyzje o wynajęciu czynników produkcji podejmowane są na nowo i przedsiębiorstwa biorą ceny i decyzje gospodarstw domowych jako dane (w szczególności nie mają wpływu na zasób kapitału), to maksymalizacja jakkolwiek zdyskontowanego strumienia zysków redukuje się do maksymalizacji zysku w każdym z okresów, a więc zachowanie przedsiębiorstwa funkcjonującego w okresie t nie zależy od tego czy istniało wcześniej i czy będzie istniało dalej. Mówiąc językiem programowania dynamicznego: w takim ujęciu przedsiębiorstwa nie mają zmiennych stanu, więc nie mogą się pojawić międzyokresowe efekty.

5 Konsumpcja i inwestycje w równowadze ogólnej

5.2.1 Wymiana w okresie $t = 0$

Załóżmy, że konsumenci i przedsiębiorstwa przystępują do wymiany w okresie 0 przed przystąpieniem do produkcji i konsumpcji. Licytator proponuje ceny konsumpcji i ceny czynników produkcji w danym okresie p_t , w_t i r_t , które wszyscy uczestnicy wymiany traktują jako dane.

Problem i -tego gospodarstwa domowego można zapisać następująco:

$$\max \sum_{t=0}^{\infty} \beta^t u(c_t^i) \quad \text{pw.} \quad \sum_{t=0}^{\infty} p_t(c_t^i + i_t^i) \leq \sum_{t=0}^{\infty} p_t(w_t + r_t k_t^i), \quad k_{t+1}^i = (1 - \delta)k_t^i + i_t^i. \quad (5.7)$$

Lewa strona nierówności określa wartość wydatków konsumpcyjnych i inwestycyjnych gospodarstwa domowego, prawa – jego dochody. Rozwiązaniem powyższego problemu jest określenie na podstawie zasobu początkowego kapitału k_0^i oraz ciągu cen p_t , w_t i r_t , wielkości konsumpcji c_t^i i inwestycji i_t^i (a więc i kapitału k_t^i dla $t > 0$) w każdym z okresów.

Równowagę w tej gospodarce możemy określić następująco: jest to taki ciąg cen $(p_t, w_t, r_t)_{t=0}^{\infty}$, takie ciągi konsumpcji, inwestycji i posiadanego przez gospodarstwa domowe kapitału $(c_t^i, i_t^i, k_t^i)_{t=0}^{\infty}$ oraz wielkości wynajętych przez przedsiębiorstwa czynników produkcji $(k_t^j, l_t^j)_{t=0}^{\infty}$, które są rozwiązaniem (przy danych cenach) problemów konsumenta i przedsiębiorstwa odpowiednio i spełniają globalne ograniczenia zasobowe (indeks i odpowiada gospodarstwom domowym, j – przedsiębiorstwom):

$$\sum_i c_t^i + i_t^i \leq \sum_j y_t^j \left(= \sum_j F(k_t^j, l_t^j) \right)$$

oraz warunki równoważenia się popytu z podażą na rynkach czynników produkcji:⁶

$$\sum_i k_t^i = \sum_j k_t^j,$$

$$\sum_i 1 = \sum_j l_t^j.$$

Możemy teraz wstępnie określić podstawowe własności równowagi. Warunki pierwszego rzędu w problemie przedsiębiorstwa są następujące:

$$F_K = r_t, \quad F_L = w_t.$$

Zauważmy, że z powyższego wynika, że w równowadze wszystkie przedsiębiorstwa będą miały tę samą wielkość, niezależnie od ewentualnego zróżnicowania gospodarstw domowych. Będziemy rozpatrywać równowagę symetryczną (identyczne gospodarstwa domowe). Dla uproszczenia rachunków przyjmujemy, że liczba gospodarstw domowych jest równa liczbie przedsiębiorstw. W symetrycznej równowadze, w której jednemu przedsiębiorstwu odpowiada jedno gospodarstwo domowe, każde przedsiębiorstwo zatrudni jednostkę pracy i kapitał o tej samej wielkości co zasób reprezentatywnego gospodarstwa domowego. Zasób reprezentatywnego gospodarstwa

⁶ Pamiętajmy, że każde gospodarstwo domowe wynajmuje jednostkę pracy.

5.2 Podstawowa wersja modelu

domowego jest równy kapitałowi na jednostkę pracy, a więc produkt reprezentatywnego przedsiębiorstwa będzie opisany funkcją produkcji w postaci intensywnej. Na podstawie (5.4) możemy wyznaczyć ceny czynników produkcji w zależności od kapitału jakim dysponuje reprezentatywne gospodarstwo:

$$r_t = f'(k_t), \quad w_t = f(k_t) - f'(k_t)k_t.$$

Zapiszmy funkcję Lagrange'a zagadnienia (5.7) (pomijamy indeksy gospodarstw domowych):

$$\begin{aligned} \mathcal{L}((c_t)_{t=0}^{\infty}, (i_t)_{t=0}^{\infty}, (k_t)_{t=1}^{\infty}) &= \sum_{t=0}^{\infty} \beta^t u(c_t) + \lambda \sum_{t=0}^{\infty} p_t (w_t + r_t k_t - c_t - i_t) \\ &+ \sum_{t=0}^{\infty} \mu_t ((1 - \delta)k_t + i_t - k_{t+1}). \end{aligned}$$

Różniczkując po c_t , i_t i k_t i przyrównując pochodne do zera otrzymujemy:

$$\frac{\partial \mathcal{L}}{\partial c_t} = 0 \Rightarrow \beta^t u'(c_t) = \lambda p_t,$$

$$\frac{\partial \mathcal{L}}{\partial i_t} = 0 \Rightarrow \mu_t = \lambda p_t$$

$$\frac{\partial \mathcal{L}}{\partial k_t} = 0 \Rightarrow \lambda p_t r_t + \mu_t (1 - \delta) = \mu_{t-1}.$$

Z ostatnich dwóch warunków wynika, że:

$$p_t (1 - \delta + r_t) = p_{t-1},$$

$$p_{t+1} (1 - \delta + r_{t+1}) = p_t.$$

Z pierwszego z warunków otrzymujemy (dzieląc stronami $\beta^t u'(c_t) = \lambda p_t$ przez $\beta^{t+1} u'(c_{t+1}) = \lambda p_{t+1}$):

$$u'(c_t) = \beta \frac{p_t}{p_{t+1}} u'(c_{t+1}).$$

Pamiętając, że $r_t = f'(k_t)$ otrzymujemy ostatecznie (pomijamy indeks gospodarstwa domowego):

$$u'(c_t) = \beta [1 - \delta + f'(k_{t+1})] u'(c_{t+1}), \quad (5.8)$$

co pokrywa się z równaniem Eulera w problemie centralnego planisty (2.21).

Jednocześnie:⁷

$$i_t = w_t + r_t k_t - c_t = f(k_t) - f'(k_t)k_t + f'(k_t)k_t - c_t = f(k_t) - c_t.$$

Ostatecznie:

$$k_{t+1} = (1 - \delta)k_t + f(k_t) - c_t. \quad (5.9)$$

Przed przystąpieniem do analizy równań (5.8) i (5.9) określmy równowagę rekursywną w opisywanej gospodarce.

⁷ Krańcowa użyteczność z konsumpcji jest dodatnia, więc konsument nigdy nie zrezygnuje z części dochodu i mamy prawo zapisać równość.

5 Konsumpcja i inwestycje w równowadze ogólnej

5.2.2 Równowaga rekursywna

Rozpatrzmy sytuację, w której na początku każdego okresu odbywa się sprzedaż dobra finalnego, którego cenę przyjmujemy jako 1⁸ oraz wynajem czynników produkcji po cenach w_t i r_t . Konsumentom nie pozwalamy nabywać i sprzedawać praw do przyszłej konsumpcji – jedyną dostępną możliwością międzyokresowej alokacji dochodów jest akumulacja kapitału (patrz dalej). Ograniczenie budżetowe gospodarstwa domowego dysponującego kapitałem k_t^i można zapisać następująco:

$$c_t^i + i_t^i \leq w_t^i + r_t k_t^i. \quad (5.10)$$

O cenach w każdym z okresów decyduje jedyna wspólna dla wszystkich zmienna stanu: poziom zagregowanego kapitału K_t . Zakładamy, że poszczególni uczestnicy rynku biorą ewolucję K jako daną (ich wpływ na nią jest minimalny). Ewolucja K będzie opisana pewnym równaniem rekursywnym:

$$K_{t+1} = g(K_t). \quad (5.11)$$

Zmiennymi stanu gospodarstwa są zatem: posiadany kapitał i całkowity zasób kapitału w gospodarce. Równanie Bellmana ma postać:⁹

$$V(k_t^i, K_t) = \max_{c_t^i, i_t^i} \{u(c_t^i) + \beta V(k_{t+1}^i, K_{t+1})\}, \quad (5.12)$$

przy czym maksymalizacja odbywa się pod warunkami (5.10), (5.5) i (5.11).

Przez równowagę rekursywną będziemy rozumieć: funkcje wartości $V(k^i, K)$ i funkcje polityki $c(k^i, K)$ i $i(k^i, K)$ będące rozwiązaniem (5.12), decyzje przedsiębiorstw o zatrudnieniu czynników produkcji będące rozwiązaniem (5.6), funkcje cen $r(K)$, $w(K)$ oraz funkcję $g(K)$ opisującą ewolucję zagregowanego kapitału spełniające warunki:

1. oczyszczania się rynku dóbr:

$$\sum_i c_t^i + i_t^i \leq \sum_j y_t^j \left(= \sum_j F(k_t^j, l_t^j) \right),$$

2. oczyszczania się rynków czynników produkcji:

$$\sum_i k_t^i = \sum_j k_t^j,$$

$$\sum_i 1 = \sum_j l_t^j,$$

3. spójności zachowań agregatów z pojedynczymi decyzjami gospodarstw domowych:

$$K_t = \sum_i k_t^i,$$

$$I_t = \sum_i i_t^i,$$

$$g(K_t) = (1 - \delta)K_t + I_t.$$

⁸ Dobro takie określa się mianem *numéraire*.

⁹ Mimo, że na tym etapie nie wykluczaliśmy jeszcze heterogeniczności gospodarstw domowych, ze względu na te same preferencje funkcja wartości i polityki będą wspólne.

5.2 Podstawowa wersja modelu

Znanymi metodami możemy z równania (5.12) wyprowadzić równanie Eulera. Z twierdzenia o obwiedni i warunków pierwszego rzędu maksymalizacji po c_t mamy (po podstawieniu $k_{t+1}^i = (1 - \delta)k_t^i + w_t + r_t k_t^i - c_t^i$):

$$\begin{aligned} V_1(k_t^i, K_t) &= \beta(1 - \delta + r_t)V_1(k_{t+1}^i, K_{t+1}), \\ u'(c_t^i) &= \beta V_1(k_{t+1}^i, K_{t+1}), \end{aligned}$$

skąd wynika:

$$u'(c_{t-1}^i) = \beta(1 - \delta + r_t)u'(c_t^i).$$

Po podstawieniu za r_t z warunków pierwszego rzędu maksymalizacji zysku przedsiębiorstw i przesunięciu indeksów do przodu ponownie uzyskujemy (5.8).

Zauważmy, że w powyższej specyfikacji równowagi rekursywnej ograniczenia narzucone na gospodarstwa domowe były silniejsze niż w przypadku wymiany w okresie $t = 0$, gdyż nie pozwalaliśmy konsumentom się zadłużać. O ile ograniczenie (5.7) oznacza, że łączna wartość wydatków nie może przewyższać wartości dochodów, o tyle (5.10) wymusza, by w każdym okresie wydatki nie przewyższały dochodów.

Rekursywnym odpowiednikiem ograniczenia budżetowego (5.7) jest połączone z warunkiem NPG międzyokresowe ograniczenie w postaci:

$$q_t a_{t+1}^i = a_t^i + w_t + r_t k_t^i - c_t^i - i_t^i, \quad (5.13)$$

gdzie a_t^i to (być może ujemne) prawa do konsumpcji w okresie t , zaś q_t określa cenę praw do jutrzejszej konsumpcji w jednostkach konsumpcji dzisiejszej. W takim sformułowaniu funkcja wartości gospodarstwa domowego ma trzy argumenty i opisana jest następującym równaniem Bellmana:

$$V(a_t^i, k_t^i, K_t) = \max_{c_t^i, i_t^i, a_{t+1}^i} \{u(c_t^i) + \beta V(a_{t+1}^i, k_{t+1}^i, K_{t+1})\}. \quad (5.14)$$

Ćwiczenie 5.3 Określ równowagę rekursywną w tej gospodarce. **Wskazówka:** cena q będzie (podobnie jak r i w) funkcją zmiennej stanu K . Pamiętaj o warunku oczyszczania się rynku kredytowego.

Jest jasne, że w równowadze symetrycznej wartość a w każdym gospodarstwie domowym będzie równa zero, a więc znów uzyskamy ten sam wynik.

Korzystając z równania (5.14) wyprowadzimy warunki pierwszego rzędu. W tym celu dokonamy następującej zamiany zmiennych: przyjmujemy, że sterowaniami są a_{t+1}^i i k_{t+1}^i . Konsumpcja jest wtedy określona wzorem:

$$c_t^i = a_t^i + w_t + r_t k_t^i - q_t a_{t+1}^i + (1 - \delta)k_t^i - k_{t+1}^i.$$

Twierdzenie o obwiedni zastosowane do (5.14) daje:

$$\begin{aligned} V_1(a_t^i, k_t^i, K_t) &= u'(c_t^i), \\ V_2(a_t^i, k_t^i, K_t) &= (1 - \delta + r_t)u'(c_t^i). \end{aligned}$$

Z warunków pierwszego rzędu maksymalizacji po prawej stronie równania Bellmana mamy:

$$q_t u'(c_t^i) = \beta V_1(a_{t+1}^i, k_{t+1}^i, K_{t+1}),$$

5 Konsumpcja i inwestycje w równowadze ogólnej

$$u'(c_t^i) = \beta V_2(a_{t+1}^i, k_{t+1}^i, K_{t+1}).$$

Po prostych przekształceniach otrzymujemy znowu równanie (5.8) oraz następujący warunek wiążący q i r :

$$q_t = \frac{1}{1 - \delta + r_{t+1}}. \quad (5.15)$$

Warunek ten stwierdza, iż w równowadze stopy zwrotu z inwestycji w kapitał i obligacje muszą być równe. (5.15) można wyprowadzić wprost korzystając z równania (5.13). Jeśli zwiększymy inwestycje o jednostkę kosztem zakupu obligacji, w kolejnym okresie będziemy mieć $(1 - \delta)$ kapitału więcej i o tyle możemy więcej skonsuować zamiast inwestować oraz uzyskamy dodatkowe r_{t+1} zwrotu z kapitału. Jednocześnie nasze prawa z tytułu obligacji zmniejszą się (w jednostkach jutrzejszej konsumpcji) o q_t^{-1} . Ponieważ łączny efekt tej operacji powinien być zerowy musi zachodzić (5.15).

5.2.3 Uwaga na temat konwencji czasowej

W powyższych rozważaniach w zupełnie naturalny sposób przyjmowaliśmy kapitał indeksowany czasem t jako czynnik produkcji w okresie t . Mimo swej naturalności podejście owo prowadzi do ustalenia stopy procentowej równej (pomniejszonej o deprecjację) krańcowej produktywności kapitału jutro, tj. w okresie $t + 1$. We współczesnej literaturze ekonomicznej coraz częściej pojawia się następująca specyfikacja funkcji produkcji:

$$Y_t = F(K_{t-1}, L_t). \quad (5.16)$$

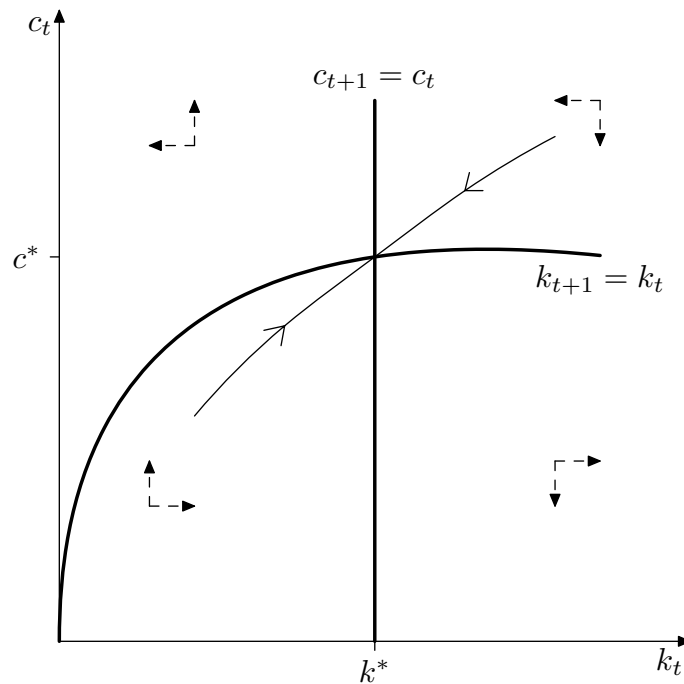
Równanie (5.16) można interpretować następująco: gospodarstwa domowe w dniu t decydują o poziomie posiadanego kapitału *na koniec dnia*. Następnego dnia rano kapitał ten jest przekazywany przedsiębiorstwom do dyspozycji i wraz z nakładem pracy w okresie $t + 1$ przyczynia się do wytworzenia produktu Y_{t+1} .

Oprócz ustalenia stopy procentowej w okresie t równej krańcowemu produktowi kapitału indeksowanego momentem t powyższe podejście pozwala ujednoczyć indeksy czasowe zmiennych decyzyjnych – dla endogenicznej zmiennej x poziom x_t jest ustalany w okresie t . Z podobnej konwencji będziemy jeszcze korzystali w późniejszych rozważaniach.

5.3 Stan ustalony

Zajmiemy się teraz analizą równań (5.8) i (5.9). Równanie (5.9) określa jutrzejszy poziom kapitału przy danym dzisiejszym jego poziomie oraz sterowaniu c_t . Zauważmy, że dla $f(k_t) - c_t > \delta k_t$ kapitał rośnie (inwestycje brutto przewyższają amortyzację), gdy $f(k_t) - c_t < \delta k_t$, kapitał maleje. Równanie Eulera (5.8) określa ewolucję optymalnej polityki. Zarówno występujące w nim c_{t+1} jak i c_t są przedmiotem wyboru, przyjmijmy jednak na chwilę, iż c_t jest określone. Wtedy warunek (5.8) określa w sposób uwikłany c_{t+1} jako funkcję c_t . Gdy $1 < \beta[1 - \delta + f'(k_{t+1})]$, konsumpcja rośnie (pamiętajmy, że $u'(c)$ jest funkcją malejącą), gdy $\beta[1 - \delta + f'(k_{t+1})] < 1$, konsumpcja maleje.

5.3 Stan ustalony



Rysunek 5.1: Diagram fazowy przestrzeni stanów w modelu Ramseya

5

Rysunek 5.1 przedstawia diagram fazowy odwzorowania $(k_{t+1}(c_t, k_t), c_{t+1}(c_t, k_t))$. Prosta $\beta[1 - \delta + f'(k)] = 1$ i krzywa $c = f(k) - \delta k$ dzielą ćwiartkę $k, c \geq 0$ na cztery części, w których c i k ewoluują w różnych (zaznaczonych na rysunku) kierunkach.

Para punktów k^*, c^* określona układem równań

$$\begin{cases} \beta[1 - \delta + f'(k^*)] = 1, \\ c^* = f(k^*) - \delta k^* \end{cases}$$

jest punktem stałym odwzorowania $(k_{t+1}(c_t, k_t), c_{t+1}(c_t, k_t))$. Parę tę nazywa się stanem ustalonym.¹⁰

Twierdzimy, że przy zadanym k_t optymalne c_t będzie tak dobrane, by gospodarka zbiegała po (zaznaczonej na rysunku) ścieżce siodłowej. Do takiego wyniku prowadzi poniższe rozumowanie: na ścieżce siodłowej spełnione jest równanie Eulera oraz warunek transwersalności ($V'(k)k$ zbiega do stałej, więc $\beta^t V'(k)k \rightarrow 0$), a zatem plan ten (ciąg sterowań c_t, c_{t+1}, \dots) jest optymalny na mocy twierdzenia 3.5. Dla pełnego formalizmu należałoby pokazać, że dla każdego k_t istnieje ścieżka spełniająca równanie Eulera i prowadząca do stanu ustalonego. Dowód ten pomijamy odsyłając zainteresowanego Czytelnika do Stokey, Lucasa i Prescottta [13] (twierdzenia 6.8 i 6.9).

Przyjrzyjmy się jeszcze równaniu określającemu kapitał w stanie ustalonym. Można ten warunek zapisać następująco:

$$f'(k^*) - \delta = \beta^{-1} - 1.$$

Jak wskazywaliśmy wcześniej, wyrażenie $\beta^{-1} - 1$ ma interpretację indywidualnej stopy dyskontującej, zaś $f'(k^*) - \delta$ jest stopą zwrotu z kapitału w stanie ustalonym.

¹⁰ Ang. *steady state*.

5 Konsumpcja i inwestycje w równowadze ogólnej

Podobnie jak w czystej wymianie, stopa procentowa (stopa zwrotu) w równowadze ustala się na poziomie odzwierciedlającym preferencje gospodarujących podmiotów. Nieco sztucznie wyglądające założenie $\beta(1+r) = 1$, które pojawia się często w modelach równowagi cząstkowej czy też problemach pojedynczych podmiotów, jest więc niczym innym jak zapisaniem wyniku równowagi ogólnej.

5.4 Egzogeniczny wzrost ludności i postęp technologiczny

Jednym z podstawowych „stylizowanych faktów” makroekonomii jest stwierdzenie, iż w długim okresie produkt (zarówno globalny jak i *per capita*) rosną w mniej więcej stałym tempie. W naturalny sposób stałe zwiększanie się produktu *per capita* interpretowane jest jako konsekwencja postępu technologicznego. Celem niniejszego podrozdziału jest rozszerzenie omawianego do tej pory modelu o wzrost ludności i egzogeniczny postęp technologiczny.

W dalszym ciągu zakładamy, iż gospodarke zaludnia wiele rodzin (gospodarstw domowych), których liczebność rośnie z okresu na okres o czynnik $1+n$. Gospodarstwo domowe uwzględnia zdyskontowaną użyteczność wszystkich członków rodziny, tak więc preferencje na strumieniu konsumpcji c_t^i reprezentatywnego (jednego) członka rodziny w okresie t określone są przez następujące wyrażenie:

$$\sum_{t=0}^{\infty} \beta^t (1+n)^t u(c_t^i). \quad (5.17)$$

Naturalnie dla zbieżności powyższej sumy musi zachodzić nierówność $\beta(1+n) < 1$. Jeśli dopuścimy wzrost gospodarczy zwiększający konsumpcję *per capita* c_t^i musimy zabezpieczyć zbieżność sumy jeszcze innym warunkiem. W szczególności, dla funkcji użyteczności chwilowej CRRA:¹¹

$$u_{\sigma}(c) = \begin{cases} \frac{c^{1-\sigma}-1}{1-\sigma} & \text{dla } \sigma > 0 \wedge \sigma \neq 1, \\ \ln c & \text{dla } \sigma = 1 \end{cases} \quad (5.18)$$

wzrost c_t^i w tempie g implikuje wzrost użyteczności w tempie¹² $(1-\sigma)g$ i warunkiem zbieżności określającej preferencje sumy jest następująca nierówność:

$$\beta(1+n)(1+(1-\sigma)g) < 1.$$

Postęp technologiczny zwiększa produkt przy tych samych nakładach pracy i kapitału. Niech A_t oznacza poziom technologii w okresie t . Są trzy metody uwzględniania A_t w funkcji produkcji:

— postęp techniczny neutralny według Hicksa zwiększający produkt A_t razy:¹³

$$Y_t = A_t F(K_t, L_t), \quad (5.19)$$

¹¹ Ang. *Constant Relative Risk Aversion*. Genezę tego określenia wyjaśnimy w części drugiej.

¹² Korzystamy z przybliżonej równości $(1+x)^n \approx 1+nx$ dla małych x .

¹³ Zauważmy, że w związku z dodatnią jednorodnością funkcji produkcji jest to równoważne z przyjęciem, iż postęp prowadzi do zwiększenia efektywnych nakładów obu czynników produkcji A_t razy, mamy bowiem $Y_t = A_t F(K_t, L_t) = Y_t = F(A_t K_t, A_t L_t)$.

5.4 Egzogeniczny wzrost ludności i postęp technologiczny

- postęp techniczny neutralny według Harroda zwiększający efektywny nakład pracy:

$$Y_t = F(K_t, A_t L_t) \quad (5.20)$$

oraz

- postęp techniczny neutralny według Solowa zwiększający efektywny nakład kapitału:

$$Y_t = F(A_t K_t, L_t). \quad (5.21)$$

Wyjaśnienie występujących powyżej określeń neutralności zawierają poniższe ćwiczenia.

Ćwiczenie 5.4 Niech $\bar{F}(K, L) = AF(K, L)$. Pokaż, że dla $K/L = \text{const}$ zachodzi $\bar{F}_K/\bar{F}_L = \text{const}$. Neutralność postępu technologicznego według Hicksa oznacza stały iloraz krańcowych produktywności przy stałym ilorazie nakładów czynników produkcji.

Ćwiczenie 5.5 Niech $\bar{F}(K, L) = F(K, AL)$. Pokaż, że dla $\bar{F}(K, L)/K = \text{const}$ zachodzi $\bar{F}_K K/(\bar{F}_L L) = \text{const}$. Neutralność postępu technologicznego według Harroda oznacza stałą proporcję wkładów czynników produkcji przy stałym ilorazie produkt/kapitał.

Ćwiczenie 5.6 Przyjmij $\bar{F}(K, L) = F(AK, L)$. Pokaż, że dla $\bar{F}(K, L)/L = \text{const}$ zachodzi $\bar{F}_K K/(\bar{F}_L L) = \text{const}$. Neutralność postępu technologicznego według Solowa oznacza stałą proporcję wkładów czynników produkcji przy stałym ilorazie produkt/praca.

Poniżej będziemy się zajmować tylko postępowaniem technicznym zasilającym pracę, tj. neutralnym według Harroda. Podstawowym powodem jest fakt, iż tylko w takim wypadku możemy uzyskać stałe stopy wzrostu wszystkich agregatów w długim okresie.¹⁴ Czytelnikowi należą się dwie uwagi: w modelach realnego cyklu koniunkturalnego poziom szoku technologicznego pojawia się przed funkcją produkcji, ponieważ szok technologiczny w modelach RBC jest stacjonarny (jego długookresowy poziom jest stały) nie jest to sprzeczne z długookresowym zachowaniem się gospodarek; w przypadku funkcji produkcji Cobba-Douglasa wszystkie trzy ujęcia postępu są równoważne.¹⁵

Będziemy przyjmować, że poziom technologii ewoluuje zgodnie z następującym równaniem:

$$A_{t+1} = (1 + g)A_t. \quad (5.22)$$

Ograniczymy się do określenia równowagi rekursywnej w przypadku, w którym jedynym aktywem jakim dysponują gospodarstwa domowe jest kapitał. Tak jak wcześniej przyjmujemy, że k_t^i oznacza kapitał na jednego członka gospodarstwa domowego. Ograniczenia, przy których maksymalizowana jest funkcja celu (5.17) są następujące:

$$(1 + n)k_{t+1}^i = (1 - \delta)k_t^i + i_t^i, \quad (5.23)$$

¹⁴ Dowód tej tezy można znaleźć u Barro i Sala-i-Martin'a [5], str. 54.

¹⁵ Niech $F(K, L) = K^\alpha L^{1-\alpha}$. Mamy $F(AK, L) = A^\alpha K^\alpha L^{1-\alpha} = A^\alpha F(K, L)$ i $F(K, AL) = A^{1-\alpha} K^\alpha L^{1-\alpha} = A^{1-\alpha} F(K, L)$. Definicje postępu technologicznego neutralnego według Harroda i Solowa są więc równoważne z definicją Hicksa, w której miernikiem poziomu technologii jest odpowiednio $A' = A^\alpha$ lub $A'' = A^{1-\alpha}$.

5 Konsumpcja i inwestycje w równowadze ogólnej

$$c_t^i + i_t^i \leq w_t + r_t k_t^i. \quad (5.24)$$

Występowanie czynnika $1 + n$ po lewej stronie równości (5.23) wynika z prostego rachunku. Niech w okresie t gospodarstwo domowe ma N_t^i członków. Wtedy łączny kapitał i inwestycje w tym gospodarstwie wynoszą $k_t^i N_t^i$ i $i_t^i N_t^i$, zaś jutrzejszy łączny kapitał jest równy $[(1 - \delta)k_t^i + i_t^i]N_t^i$. Liczba członków gospodarstwa w następnym okresie wzrośnie do $N_{t+1}^i = (1 + n)N_t^i$, skąd mamy:

$$k_{t+1}^i = \frac{[(1 - \delta)k_t^i + i_t^i]N_t^i}{N_{t+1}^i} = \frac{(1 - \delta)k_t^i + i_t^i}{1 + n}.$$

Zmiennymi stanu gospodarstwa są: posiadany kapitał, całkowity zasób kapitału w gospodarce oraz poziom technologii A_t . Równanie Bellmana ma postać:

$$V(k_t^i, K_t, A_t) = \max_{c_t^i, i_t^i} \{u(c_t^i) + \beta(1 + n)V(k_{t+1}^i, K_{t+1}, A_{t+1})\}. \quad (5.25)$$

Problem przedsiębiorstw jest następujący:

$$\max \Pi_t^j = F(k_t^j, A_t l_t^j) - w_t l_t^j - r_t k_t^j. \quad (5.26)$$

Na równowagę składają się: funkcja wartości $V(k^i, K, A)$ i polityki $c(k^i, K, A)$ i $i(k^i, K, A)$ będące rozwiązaniem (5.25), decyzje przedsiębiorstw o zatrudnieniu czynników produkcji będące rozwiązaniem (5.26), funkcje cen $r(K, A)$, $w(K, A)$ oraz funkcja ewolucji kapitału $K_{t+1} = g(K_t)$ spełniająca warunki oczyszczania się rynków dóbr i czynników produkcji oraz warunki spójności zachowań agregatów z pojedynczymi decyzjami gospodarstw domowych.

Możemy przejść do określenia warunków pierwszego rzędu przedsiębiorstw i gospodarstw domowych. Ze względu na fakt, iż interesuje nas równowaga symetryczna, w dalszych rozważaniach pomijamy indeksy gospodarstw domowych i przedsiębiorstw. Dokonajmy podstawienia:

$$k_{t+1} = \frac{(1 - \delta)k_t + w_t + r_t k_t - c_t}{1 + n}.$$

Z twierdzenia o obwiedni i warunku pierwszego rzędu maksymalizacji w równaniu Bellmana mamy:

$$V_1(k_t, K_t, A_t) = \beta(1 + n) \frac{(1 - \delta + r_t)}{1 + n} V_1(k_{t+1}, K_{t+1}, A_{t+1})$$

oraz:

$$u'(c_t) = \beta(1 + n) \frac{1}{1 + n} V_1(k_{t+1}, K_{t+1}, A_{t+1}).$$

Z powyższych równań uzyskujemy znajomą postać równania Eulera:

$$u'(c_t) = \beta(1 - \delta + r_t)u'(c_{t+1}).$$

Przed przystąpieniem do dalszych obliczeń dokonamy następujących podstawień:

$$c_t^* = \frac{c_t}{A_t}, \quad i_t^* = \frac{i_t}{A_t}, \quad k_t^* = \frac{k_t}{A_t}, \quad w_t^* = \frac{w_t}{A_t}, \quad r_t^* = r_t.$$

5.4 Egzogeniczny wzrost ludności i postęp technologiczny

c^* , i^* i k^* oznaczają konsumpcję, inwestycje i kapitał na jednostkę efektywnej pracy, zaś w^* to wynagrodzenie jednostki efektywnej pracy. Korzystając z powyższych podstawień problem maksymalizacji zysku przedsiębiorstwa możemy przepisać następująco:¹⁶

$$\max \Pi_t = A_t [F(k_t^*, l_t) - w_t^* l_t - r_t^* k_t^*].$$

Warunki pierwszego rzędu mają postać (korzystamy z (5.4)):

$$r_t^* = f'(k_t^*), \quad w_t^* = f(k_t^*) - f'(k_t^*)k_t^*.$$

Przyjmijmy funkcję użyteczności chwilowej postaci CRRA. Korzystając z faktu, iż $u'(c) = c^{-\sigma}$, możemy przepisać równanie Eulera następująco:

$$u'(c_t^*) = \frac{\beta(1 - \delta + f'(k_t^*))}{(1 + g)^\sigma} u'(c_{t+1}^*). \quad (5.27)$$

Po odpowiednich podstawieniach uzyskujemy równanie opisujące ewolucję kapitału na jednostkę efektywnej pracy:

$$k_{t+1}^* = \frac{(1 - \delta)k_t^* + f(k_t^*) - c_t^*}{(1 + n)(1 + g)}. \quad (5.28)$$

Widoczne jest podobieństwo układu równań (5.27)-(5.28) i układu (5.8)-(5.9). Analiza diagramu fazowego pozostaje bez zmian. Stan ustalony spełnia układ równań:

$$\begin{cases} \beta[1 - \delta + f'(\bar{k})] = (1 + g)^\sigma, \\ \bar{c} = f(\bar{k}) - (\delta + n + g + ng)\bar{k}. \end{cases}$$

Zauważmy, iż w tak określonym stanie ustalonym kapitał i produkt *per capita* rosną w tempie g zaś agregaty w tempie $(1 + g)(1 + n) - 1 = n + g + ng$.

Czy jesteśmy w stanie zagwarantować zbieżność do stanu ustalonego \bar{k} , \bar{c} ? Znów odwołajmy się do warunku transwersalności. Zachodzą poniższe równości:

$$\begin{aligned} \beta^t (1 + n)^t V_1(k_t, K_t, A_t) k_t &= \beta^t (1 + n)^t u'(c_{t-1}) k_t = \beta^t (1 + n)^t u'(c_{t-1}^* A_t) k_t^* A_t = \\ &= \beta^t (1 + n)^t u'(c_{t-1}^*) k_t^* A_t^{1-\sigma} = \beta^t (1 + n)^t u'(c_{t-1}^*) k_t^* A_0^{1-\sigma} (1 + g)^{(1-\sigma)t}. \end{aligned}$$

Ponieważ

$$u'(c_{t-1}^*) k_t^* A_0^{1-\sigma} \rightarrow \text{const},$$

¹⁶ Czytelnikowi należy się tutaj jedna uwaga. Poniższy zapis oznacza, iż opuszczając indeksy gospodarstw domowych i przedsiębiorstw *implicite* przyjęliśmy, że w każdym okresie jednemu członkowi gospodarstwa domowego odpowiada jedno przedsiębiorstwo (z okresu na okres liczba przedsiębiorstw zwiększa się o czynnik $1 + n$) i będzie $l_t = 1$. Alternatywne podejście polegałoby na ustaleniu stałej liczby przedsiębiorstw. Niech L_t oznacza liczbę ludności w chwili t , J – liczbę przedsiębiorstw. Oznaczyliśmy przez k_t kapitał na jednego członka gospodarstwa domowego (równy kapitałowi na mieszkańca K/L_t) i przez k_t^* – kapitał na jednostkę efektywnej pracy w gospodarstwie domowym (równy kapitałowi na „efektywnego” mieszkańca $K/(AL_t)$). Przyjmując stałą liczbę przedsiębiorstw musielibyśmy przyjąć dodatkowe oznaczenie na kapitał na jednostkę efektywnej zatrudnianej w nich pracy, które występowałoby we wszystkich wzorach dotyczących przedsiębiorstw. Zatrudnienie w przedsiębiorstwie wynosiłoby $l_t = L_t/J$ i kapitał na jednostkę efektywnej pracy w przedsiębiorstwie równałby się $KJ/(AL_t)$. Oczywiście wyniki naszych rozważań pozostałyby niezmiennione.

5 Konsumpcja i inwestycje w równowadze ogólnej

więc dla spełnienia warunku transwersalności wystarczy zapewnić zbieżność:

$$\beta^t(1+n)^t(1+g)^{(1-\sigma)t} \rightarrow 0,$$

która ma miejsce wtedy i tylko wtedy gdy:

$$\beta(1+n)(1+g)^{1-\sigma} \approx \beta(1+n)(1+(1-\sigma)g) < 1.$$

Jak wcześniej wskazywaliśmy, powyższy warunek jest warunkiem koniecznym zbieżności sumy określającej użyteczność. Wnioskujemy więc, iż nasza gospodarka zbiega do ścieżki zrównoważonego wzrostu, na której wszystkie agregaty rosną w stałym tempie.

Z warunku określającego stan ustalony wynika, iż stopa zwrotu z kapitału (a zatem i stopa procentowa) ustali się w długim okresie na poziomie:

$$f'(\bar{k}) - \delta = (1+g)^\sigma \beta^{-1} - 1.$$

Pamiętając określenie $\beta = (1+\theta)^{-1}$, gdzie θ to stopa preferencji czasowej mamy (patrz też zadanie 4.1):

$$f'(\bar{k}) - \delta \approx \sigma g + \theta. \quad (5.29)$$

5.5 Wprowadzenie do wyceny aktywów

Zrezygnujemy teraz ze sztucznego założenia, iż gospodarstwa domowe są bezpośrednio właścicielami kapitału, który wypożyczają powstającym na jeden okres przedsiębiorstwom. Przyjmujemy następującą strukturę własności: gospodarstwa domowe dysponują udziałami Z w przedsiębiorstwach, które z kolei są właścicielami kapitału i podejmują decyzje inwestycyjne. Dla uproszczenia rachunków zakładamy brak postępu technologicznego i stałą liczbę ludności. Przyjmujemy konwencję czasową sygnalizowaną na stronie 79, tak więc Z_{t-1}^i oznaczają udziały posiadane na początku okresu t (wykupione na koniec okresu $t-1$), zaś k_{t-1}^j oznacza kapitał określony w okresie $t-1$ używany do produkcji w okresie t .

Ograniczymy się do analizy wymiany w okresie $t=0$. Niech p_t będzie ceną produktu wytworzonego (konsumpcji) w okresie t . Celem przedsiębiorstwa jest maksymalizacja jego wartości:

$$V^j = \sum_{t=0}^{\infty} p_t \Pi_t^j, \quad (5.30)$$

gdzie zysk Π_t to:

$$\Pi_t^j = F(k_{t-1}^j, l_t^j) - w_t l_t^j - i_t^j. \quad (5.31)$$

Kapitał jakim dysponuje przedsiębiorstwo ewoluuje następująco:

$$k_t^j = (1-\delta)k_{t-1}^j + i_t^j. \quad (5.32)$$

Przedsiębiorstwa wypłacają wygenerowane zyski w formie dywidendy.

5.5 Wprowadzenie do wyceny aktywów

Gospodarstwo domowe czerpie przychody z pracy w_t oraz z dywidend d_t na jedną akcję. Dodatkowo P_t jest ceną jednej akcji, którą gospodarstwo może sprzedać. Gospodarstwo domowe przeznaczają swoje dochody na wydatki konsumpcyjne i zakup akcji na następny okres. Ograniczenie budżetowe ma postać:

$$\sum_{t=0}^{\infty} p_t(c_t^i + P_t Z_t^i) \leq \sum_{t=0}^{\infty} p_t(w_t + Z_{t-1}^i(P_t + d_t)). \quad (5.33)$$

Przyjmujemy, że na każde gospodarstwo przypada jedno przedsiębiorstwo i początkowo gospodarstwo domowe dysponuje jedną akcją $Z_{-1} = 1$.

Równowagę stanowią: ceny $(p_t)_{t=0}^{\infty}, (P_t)_{t=0}^{\infty}, (w_t)_{t=0}^{\infty}$ oraz ilości $(c_t^i)_{t=0}^{\infty}, (Z_t^i)_{t=0}^{\infty}, (k_t^j)_{t=0}^{\infty}, (i_t^j)_{t=0}^{\infty}, (d_t^j)_{t=0}^{\infty}$ będące rozwiązaniami problemów przedsiębiorstw i gospodarstw domowych przy tych cenach, które spełniają globalne ograniczenia zasobów oraz warunki oczyszczania się rynku pracy i rynku akcji. W równowadze symetrycznej (każde przedsiębiorstwo dysponuje tym samym początkowym zasobem kapitału), wartości i wypłacane dywidendy wszystkich przedsiębiorstw będą identyczne i otrzymamy $Z_t = 1$ dla każdego gospodarstwa.

Przejdźmy do określenia podstawowych cech równowagi. Pomijamy indeksy gospodarstw domowych i przedsiębiorstw. Funkcja Lagrange'a problemu maksymalizacji użyteczności gospodarstw domowych pod warunkiem (5.33) ma następującą postać:

$$\mathcal{L} = \sum_{t=0}^{\infty} \beta^t u(c_t) + \lambda \sum_{t=0}^{\infty} p_t[w_t + Z_{t-1}(P_t + d_t) - c_t - P_t Z_t]$$

Warunki pierwszego rzędu to:

$$\frac{\partial \mathcal{L}}{\partial c_t} = 0 \Rightarrow \beta^t u'(c_t) = \lambda p_t,$$

$$\frac{\partial \mathcal{L}}{\partial Z_t} = 0 \Rightarrow \lambda p_t P_t = \lambda p_{t+1}(P_{t+1} + d_{t+1}),$$

skąd otrzymujemy:

$$P_t = \frac{p_{t+1}}{p_t}(P_{t+1} + d_{t+1}). \quad (5.34)$$

Dzisiejsza cena akcji przedsiębiorstwa musi być zatem równa zdyskontowanej (iloraz p_{t+1}/p_t określa dzisiejszą wartość jutrzejszej konsumpcji) jutrzejszej cenie i dywidendzie.

Warunek (5.34) ma naturalną interpretację jako warunek braku możliwości arbitrażu między rynkiem kredytowym¹⁷ a rynkiem akcji: rezygnacja z jednostki konsumpcji w okresie t pozwala zakupić p_t/p_{t+1} jednostek konsumpcji jutro; jednocześnie rezygnując z jednostki konsumpcji gospodarstwo może zakupić P_t^{-1} akcji, które jutro dadzą wypłatę $P_t^{-1}(P_{t+1} + d_{t+1})$. Przyrównanie (mierzonych jutrzejszą konsumpcją) wypłat z obu strategii daje (5.34).

Równanie (5.34) można rozwiązywać rekursywnie uzyskując:

$$P_t = \frac{p_{t+1}}{p_t}(P_{t+1} + d_{t+1}) =$$

¹⁷ Możliwość międzyokresowej realokacji konsumpcji przy ustalonej ścieżce dochodów zapisana jest w ograniczeniu budżetowym (5.33).

5 Konsumpcja i inwestycje w równowadze ogólnej

$$\frac{p_{t+1}}{p_t} \left(\frac{p_{t+2}}{p_{t+1}} (P_{t+2} + d_{t+2}) + d_{t+1} \right) = \frac{p_{t+1}}{p_t} d_{t+1} + \frac{p_{t+2}}{p_t} d_{t+2} + \frac{p_{t+2}}{p_t} P_{t+2}.$$

Dalsze podstawienia prowadzą do następującej wyceny akcji:

$$P_t = \sum_{\tau=1}^T \frac{p_{t+\tau}}{p_t} d_{t+\tau} + \frac{p_{t+T}}{p_t} P_{t+T}. \quad (5.35)$$

Jeśli:

$$\lim_{T \rightarrow \infty} \frac{p_{t+T}}{p_t} P_{t+T} = 0, \quad (5.36)$$

wtedy możemy określić wartość przedsiębiorstwa jako sumę zdyskontowanych dywidend od „jutra” do nieskończoności:

$$P_t = \sum_{\tau=1}^{\infty} \frac{p_{t+\tau}}{p_t} d_{t+\tau}. \quad (5.37)$$

Ćwiczenie 5.7 W równaniu (5.34) dokonaj podstawienia $1 + r_t = p_t/p_{t+1}$. Znajdź odpowiedniki równań (5.35)-(5.37).

Warunek (5.36) wyklucza tzw. bąble spekulacyjne zrównując wycenę akcji z jej fundamentalną wartością określoną przez (5.37). Niech P_t^* będzie wartością fundamentalną akcji. Będziemy poszukiwać ogólnych rozwiązań równania (5.34) w postaci $P_t^* + B_t$. Mamy:

$$P_t^* + B_t = \frac{p_{t+1}}{p_t} (P_{t+1}^* + B_{t+1} + d_{t+1}).$$

Korzystając z faktu, iż samo P_t^* spełnia (5.34) uzyskujemy równanie określające ewolucję B_t :

$$B_{t+1} = \frac{p_t}{p_{t+1}} B_t.$$

Bąbel musi więc rosnąć w tempie wyznaczonym przez stopę procentową. Jeśli dziś na rynku akcji cena przewyższa wartość fundamentalną o B_t , to jutro musi ją przewyższać o B_{t+1} – bąbel jest wynikiem samospełniającej się przepowiedni. Zauważmy, że ponieważ bąbel rośnie szybciej niż gospodarka¹⁸ w pewnym momencie jedna akcja musiałaby być warta więcej niż wynosi zagregowany produkt.

Na koniec naszych rozważań określimy, jak będzie wyglądała polityka inwestycyjna przedsiębiorstw. Lagranżan w problemie maksymalizacji zysku przedsiębiorstwa ma postać:

$$\mathcal{L} = \sum_{t=0}^{\infty} p_t [f(k_{t-1}) - w_t - i_t] + \sum_{t=0}^{\infty} \lambda_t [(1 - \delta)k_{t-1} + i_t - k_t].$$

Różniczkowanie po i_t oraz k_t i przyrównanie pochodnych do zera daje:

$$\frac{\partial \mathcal{L}}{\partial i_t} = -p_t + \lambda_t,$$

¹⁸ W modelu bez wzrostu PKB w długim okresie jest stałe. W modelu ze wzrostem stopa procentowa przewyższa tempo wzrostu gospodarki, poza przypadkami szczególnymi, np. $\sigma < 1 - \theta/g$ w równaniu (5.29).

Zadania

$$\frac{\partial \mathcal{L}}{\partial k_t} = p_{t+1} f'(k_t) + \lambda_{t+1}(1 - \delta) - \lambda_t,$$

skąd:

$$p_{t+1}[1 - \delta + f'(k_t)] = p_t.$$

Zauważmy, że podstawienie za p_t z warunków pierwszego rzędu konsumenta daje:

$$\frac{\beta^{t+1} u'(c_{t+1})}{\lambda} [1 - \delta + f'(k_t)] = \frac{\beta^t u'(c_t)}{\lambda},$$

co redukuje się do równania:

$$u'(c_t) = \beta u'(c_{t+1}) [1 - \delta + f'(k_t)]. \quad (5.38)$$

Ponieważ $\Pi_t = d_t$ odpowiednie przekształcenia równań określających zysk i ewolucję kapitału dają

$$k_t = (1 - \delta)k_{t-1} + f(k_{t-1}) - d_t - w_t.$$

W równowadze gospodarstwo domowe będzie konsumowało w każdym okresie swe przychody z pracy i dywidend (portfel akcji nie będzie się zmieniał), a więc:

$$d_t + w_t = c_t$$

i

$$k_t = (1 - \delta)k_{t-1} + f(k_{t-1}) - c_t. \quad (5.39)$$

Nietrudno zauważyć, iż polityka inwestycyjna przedsiębiorstw opisana równaniami (5.38) i (5.39) pokrywa się z polityką, jaką prowadziłyby gospodarstwa domowe, gdyby były bezpośrednimi właścicielami kapitału i polityką, jaką prowadziłyby centralny planista.

Zadania

Zadanie 5.1 Załóżmy, że gospodarstwa domowe dzielą swój zasób czasu równy 1 w każdym z okresów między pracę l_t i przynoszący użyteczność czas wolny $1 - l_t$. Preferencje gospodarstwa domowego określone są przez następujące wyrażenie:

$$\sum_{t=0}^{\infty} \beta^t u(c_t, 1 - l_t).$$

1. Zapisz ograniczenie budżetowe gospodarstwa w przypadku wymiany w okresie $t = 0$.
2. Określ równowagę Arrowa-Debreu.
3. Zapisz ograniczenie budżetowe oraz równanie Bellmana w wymianie następującej co okres.
4. Określ równowagę rekursywną.
5. Znajdź warunki pierwszego rzędu: równanie Eulera wiążące konsumpcję dziś i jutro oraz wewnątrzokresowy warunek wiążący konsumpcję i podaż pracy.

5 Konsumpcja i inwestycje w równowadze ogólnej

6. Znajdź warunki określające stan ustalony.

Zadanie 5.2 Rozpatrz model z podrozdziału 5.5 w przypadku, w którym wymiana odbywa się na początku każdego okresu. Dopuszczaj możliwość zadłużania się przez gospodarstwa domowe. Napisz równania Bellmana konsumentów i przedsiębiorstw. Określ równowagę rekursywną. Znajdź odpowiedniki równań (5.34)-(5.37).

Rozdział 6

Metoda kolokacji

W rozdziale tym prezentujemy metodę kolokacji i jej zastosowanie w problemach programowania dynamicznego z ciągłą przestrzenią stanów i ciągłym sterowaniem. Metoda kolokacji jest szczególnym przypadkiem tzw. metod projekcyjnych,¹ jest jednak spośród nich najprostsza w implementacji i zapewnia bardzo dobre rezultaty. Metody projekcyjne aproksymują szukaną funkcję globalnie (na pewnym przedziale) i są szczególnie przydatne w zastosowaniu do zagadnień wzrostu, gdzie interesuje nas zachowanie gospodarki daleko od stanu ustalonego. Numeryczne ograniczenia pozwalają na skuteczne stosowanie metod projekcyjnych tylko w wypadku kilku zmiennych stanu. W modelach cyklu koniunkturalnego, gdzie kluczowe dla analizy jest zachowanie wokół stanu ustalonego, lepsze są metody lokalne prezentowane w części trzeciej.

Przed lekturą tego rozdziału zalecane jest zapoznanie się z dodatkiem B, gdzie opisane są wielomiany Czebyszewa i zakodowane funkcje MATLAB/OCTAVE wykorzystywane poniżej.

6.1 Metody projekcyjne i metoda kolokacji

Rozpatrzmy problem znalezienia funkcji $f: [a, b] \rightarrow \mathbb{R}$ spełniającej równanie funkcyjne:

$$T(f)(x) = 0. \quad (6.1)$$

Bardzo wiele zagadnień da się zapisać w postaci (6.1). Może to być np. równanie różniczkowe, równanie Bellmana czy równanie określające funkcję odwrotną do pewnej funkcji.

Żałóżmy, że chcemy przybliżyć f funkcją \bar{f} określoną wzorem:

$$\bar{f}(x) = \sum_{n=0}^{N-1} a_n \phi_n(x), \quad (6.2)$$

¹ Patrz Judd [11], rozdział 7.

6 Metoda kolokacji

gdzie funkcje $\phi_n(x)$ należą do pewnej rodziny, np. wielomianów x^n czy wielomianów Czebyszewa $T_n(x)$. W ogólności chcemy dla danego N wyznaczyć takie współczynniki a_0, \dots, a_{N-1} aby \bar{f} najlepiej przybliżała nieznaną funkcję f z równania (6.1). Nazwa „metody projekcyjne” bierze się właśnie stąd: „rzutujemy” nieznaną rozwiązanie na skończeniowym wymiarową przestrzeń funkcji $\sum_{n=0}^{N-1} a_n \phi_n(x)$.

Jak poszukiwać owego „najlepszego” przybliżenia? Zdefiniujmy funkcję błędu $R(x; a_0, \dots, a_{N-1})$ następująco:

$$R(x; a_0, \dots, a_{N-1}) = T(\bar{f})(x). \quad (6.3)$$

Funkcja R mówi nam jak bardzo \bar{f} odbiega od dokładnego rozwiązania (6.1) w punkcie x .

Mając funkcję R możemy określić wiele różnych kryteriów wyboru wektora wag $(a_n)_{n=0}^{N-1}$. Może to być na przykład minimalizacja błędu średniokwadratowego:

$$\min_{a_0, \dots, a_{N-1}} \int_a^b R(x; a_0, \dots, a_{N-1})^2 dx. \quad (6.4)$$

Innym kryterium, w przypadku gdy $\phi_n(x) = x^n$, jest kryterium Galerkinia:²

$$\forall i \in \{0, \dots, N-1\} \int_a^b R(x; a_0, \dots, a_{N-1}) x^i dx = 0. \quad (6.5)$$

Wymienione kryteria doboru $(a_n)_{n=0}^{N-1}$ wymagają całkowania funkcji R . Metoda kolokacji omija powyższy problem poprzez żądanie aby równanie (6.1) było dokładnie spełnione w N punktach x_1, \dots, x_N . W metodzie kolokacji $(a_n)_{n=0}^{N-1}$ są wyznaczone przez następujący układ równań nieliniowych:

$$\begin{cases} R(x_1; a_0, \dots, a_{N-1}) = 0 \\ R(x_2; a_0, \dots, a_{N-1}) = 0 \\ \vdots \\ R(x_N; a_0, \dots, a_{N-1}) = 0 \end{cases} \quad (6.6)$$

Najlepszym podejściem jest wybranie x_1, \dots, x_N tak, by były węzłami Czebyszewa na $[a, b]$ i ustalenie funkcji bazowych $\phi_n(x)$ wielomianami Czebyszewa. Taką metodę nazywa się kolokacją Czebyszewa. W dalszym ciągu będziemy używać wyłącznie tej metody.

6.2 Przykład zastosowania metody kolokacji

Pokażemy teraz jak zastosować metodę kolokacji do przybliżenia funkcji $\arcsin y$ na przedziale $[-1, 1]$. Funkcja ta określona jest poniższym równaniem:

$$\sin f(x) = x,$$

które można przepisać następująco:

$$\sin f(x) - x = 0. \quad (6.7)$$

Zacniemy od zakodowania funkcji MATLAB/OCTAVE zwracającej resztę równania (6.7) dla podanych x_i , i $f(x_i)$.

² Statystycznym odpowiednikiem metody Galerkinia jest metoda momentów.

6.2 Przykład zastosowania metody kolokacji

```

Plik eqasin.m
function y=eqasin(x, f)
y=sin(f)-x;
Koniec pliku eqasin.m

```

Korzystając z funkcji `eqasin` możemy teraz zapisać funkcję zwracającą dla zadanych współczynników a_0, \dots, a_{N-1} wektor reszt $R(x_i; a_0, \dots, a_{N-1})$ w węzłach kolokacji. Przyjmujemy liczbę węzłów równą 10. Do wyznaczenia węzłów Czebyszewa na $[-1, 1]$ korzystamy z funkcji `chebnod`. Dodatkowo korzystamy z tego, że jeśli x jest wektorem węzłów Czebyszewa, wtedy wektor wartości funkcji $f(x_i)$ przybliżanej szeregiem Czebyszewa (wzór (B-14)) z parametrami α_i jest równy $\Phi\alpha$, gdzie $\Phi_{i,j} = T_{j-1}(x_i)$. Do wyznaczenia macierzy Φ korzystamy z funkcji `chebphi`.

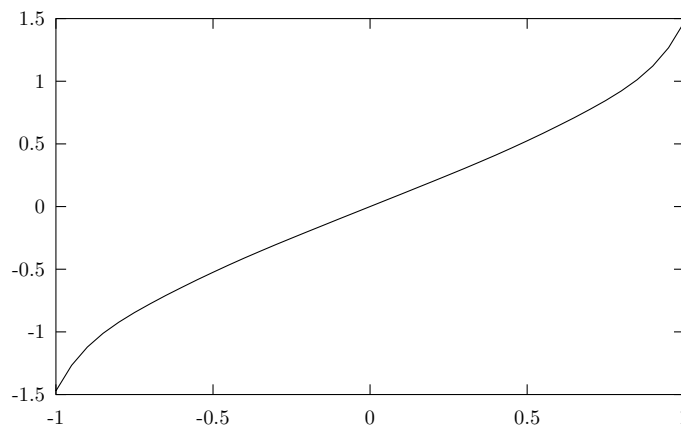
```

Plik residasin.m
function ret=residasin(v)
ret=eqasin(chebnod(10), chebphi(10)*v);
Koniec pliku residasin.m

```

Mając funkcję `residasin` możemy wyznaczyć wektor rozwiązań równania (6.6) w naszym problemie korzystając np. z metody Newtona. Jako pierwsze przybliżenie przyjmujemy wektor współczynników szeregu Czebyszewa odpowiadający funkcji $f(x) = x$.

```
octave:1> a=newton('residasin', [0 1 0 0 0 0 0 0 0 0]');
```



Rysunek 6.1: Rozwiązanie równania $\sin(f(x)) = x$ uzyskane metodą kolokacji

Mając wektor współczynników a możemy (korzystając z funkcji `chebval`) określić wartości naszego szeregu aproksymującego na całym przedziale i sporządzić wykres naszego przybliżenia (patrz rys. 6.1).

```

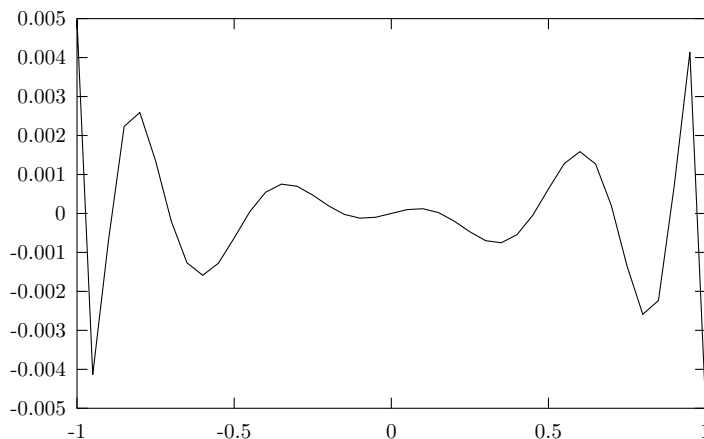
octave:2> x=(-1:.05:1)';
octave:3> y=chebval(a, x);
octave:4> plot(x,y);

```

6 Metoda kolokacji

Bardzo ważne jest sprawdzenie jak zachowują się reszty równania poza węzłami kolokacji. To, że w tych węzłach z definicji błąd będzie równy zero nie gwarantuje, iż poza nimi będzie satysfakcjonująco mały. Poniższe komendy korzystające z funkcji `eqasin` pozwalają uzyskać wykres błędu na całym przedziale (rys. 6.2).

```
octave:5> R=eqasin(x,y);
octave:6> plot(x,R);
```



Rysunek 6.2: Reszta $R(x; a_0, \dots, a_{N-1})$ w problemie $\sin(f(x)) = x$

6

Zauważmy, że błąd przybliżenia wyraźnie rośnie na krańcach przedziału. Jest to spowodowane tym, że wszystkie węzły Czebyszewa leżą w środku przedziału, a pochodna naszej przybliżanej funkcji ($[\arcsin x]' = 1/\sqrt{1-x^2}$) zbiega do $\pm\infty$, gdy zbliżamy się do ∓ 1 .

Należy zwrócić szczególną uwagę na dobór punktu startowego i analizę wyniku (zachowania się reszt). W ogólności równanie (6.6) ma więcej niż jedno rozwiązanie i to, czy nasz program do rozwiązywania równań nieliniowych zbiegnie (jeśli w ogóle) do poprawnego rozwiązania, zależy od wyboru punktu startowego.

Ćwiczenie 6.1 Znajdź zera funkcji `residasin` korzystając z funkcji `newton` przyjmując jako punkt startowy wektor współczynników szeregu Czebyszewa odpowiadający funkcjom $f(x) = -2$ i $f(x) = 10x$. Sporządź dla uzyskanych rozwiązań wykresy szeregu aproksymującego i reszt.

Ćwiczenie 6.2 Niech $D(p) = p^{-\frac{1}{2}} + 3p^{-\frac{1}{3}}$ będzie funkcją popytu. Znajdź używając metody kolokacji funkcję odwróconego popytu $p(q)$ (spełniającą $D(p(q)) = q$).

6.3 Równanie Eulera jako równanie funkcyjne

Najwyższy czas pokazać użyteczność (z punktu widzenia programowania dynamicznego) rozwiniętych do tej pory narzędzi. Kluczową rolę odgrywa tutaj nadanie nowej interpretacji równaniom Eulera – w modelach z nieskończonym horyzontem czasowym są one nie tylko nieliniowymi równaniami rekurencyjnymi opisującymi

6.4 Numeryczne rozwiązanie modelu Ramseya

optymalną politykę, są także równaniami *funkcyjnymi*, a więc można zastosować do nich metody numeryczne dla problemów funkcjonalnych, w tym metodę kolokacji.

Równanie Eulera, czy też szerzej – zestaw warunków pierwszego rzędu, można w ogólności zapisać następująco:

$$F(c_t, s_t, c_{t+1}, s_{t+1}) = 0, \quad (6.8)$$

przy czym $s_{t+1} = g(s_t, c_t)$. W wielowymiarowym przypadku n zmiennych stanu, m sterowań i k warunków pierwszego rzędu mamy: $g: \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$, $c: \mathbb{R}^n \rightarrow \mathbb{R}^m$ i $F: \mathbb{R}^m \times \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n \rightarrow \mathbb{R}^k$.

W rozdziale 2 stwierdziliśmy, że w modelach z nieskończonym horyzontem jest jedna, niezależna od czasu funkcja polityki $c(s)$. Wykorzystując ten fakt możemy równanie (6.8) zapisać następująco:

$$F(c(s_t), s_t, c(g(c(s_t), s_t)), g(s_t, c(s_t))) = 0. \quad (6.9)$$

W powyższym równaniu F i g są określone, zaś $c(\cdot)$ jest poszukiwaną przez nas funkcją polityki. W oczywisty sposób równanie (6.9) jest szczególnym przypadkiem zagadnienia (6.1) i można do jego numerycznego rozwiązania zastosować metodę kolokacji.³

6.4 Numeryczne rozwiązanie modelu Ramseya

Punktem wyjścia będzie dla nas równanie Eulera (5.8):

$$u'(c_t) = \beta[1 - \delta + f'(k_{t+1})]u'(c_{t+1}),$$

które zgodnie z wcześniejszą dyskusją można przepisać jako równanie funkcyjne następująco:

$$\beta[1 - \delta + f'(g(k_t, c(k_t)))]u'(c(g(k_t, c(k_t)))) - u'(c(k_t)) = 0, \quad (6.10)$$

gdzie:

$$g(k_t, c_t) = (1 - \delta)k_t + f(k_t) - c_t.$$

Numeryczne rozwiązanie (6.10) wymaga zakodowania najpierw funkcji użyteczności i produkcji (i ich pochodnych) oraz funkcji przejścia g .

Przyjmujemy funkcję użyteczności postaci CRRA (ang. *Constant Relative Risk Aversion*) $u_\theta(c) = c^{1-\theta}/(1-\theta)$ z parametrem $\theta = 2$. Funkcja `utilramsey` jako drugi wynik podaje krańcową użyteczność ($u'(c)$).

³ Równaniami funkcyjnymi są też w oczywisty sposób równania Bellmana. Szerokie omówienie zastosowania metody kolokacji do równań Bellmana znajdzie Czytelnik w podręczniku Mirandy i Facklera [16]. W przypadku aproksymacji funkcji wartości współczynniki kolokacji uzyskuje się iterując równanie Bellmana. Nasze podejście ma następujące zalety: jest łatwiejsze w implementacji (jeśli dysponujemy funkcją rozwiązującą układy równań nieliniowych), funkcję polityki uzyskujemy od razu i, co najważniejsze, operowanie na warunkach pierwszego rzędu pozwala bez przeszkód analizować modele, w których równowaga konkurencyjna prowadzi do alokacji nieefektywnych w sensie Pareto – analiza równań Bellmana ogranicza nas właściwie do problemów centralnego planisty (ogólniej: zagadnień optymalizacyjnych pojedynczego podmiotu).

6 Metoda kolokacji

_____ Plik *utilramsey.m* _____

```
function [u, u1]=utilramsey(c)

theta=2;

u=c.^(1-theta)/(1-theta);
u1=c.^(-theta);
```

_____ Koniec pliku *utilramsey.m* _____

Poniżej prezentujemy kod funkcji *prodframsey*, która zwraca $f(k)$ i $f'(k)$. Przyjeliśmy, że funkcja produkcji jest funkcją Cobba-Douglasa z $\alpha = 1/3$.

_____ Plik *prodframsey.m* _____

```
function [f, f1]=prodframsey(k)

alpha=1/3;

f=k.^alpha;
f1=alpha*k.^(alpha-1);
```

_____ Koniec pliku *prodframsey.m* _____

Funkcja *transramsey* określa jutrzejszy poziom kapitału dla podanego poziomu dzisiejszego kapitału i dzisiejszej konsumpcji. Jako drugi wynik *transramsey* podaje $1 - \delta + f'(k)$ (równe $g_1(k, c)$), co przyda się w kodowaniu równania Eulera, w którym występuje wyrażenie $1 - \delta + f'(k_{t+1})$. Stopę deprecjacji ustalamy na poziomie $\delta = 5\%$.

_____ Plik *transramsey.m* _____

```
function [g, g1]=transramsey(k,c)

delta=.05;

[f, f1]=prodframsey(k);

g=(1-delta)*k+f-c;
g1=(1-delta)+f1;
```

_____ Koniec pliku *transramsey.m* _____

Mając powyższe funkcje możemy teraz zakodować funkcję *eqeulerramsey* podającą resztę równania Eulera dla zadanych $k_t, c_t, k_{t+1}, c_{t+1}$. W funkcji tej pojawia się współczynnik dyskontujący β , który ustalamy na poziomie odpowiadającym stopie dyskontującej 4%.

_____ Plik *eqeulerramsey.m* _____

```
function y=eqeulerramsey(k1, c1, k2, c2)

discount=1/1.04;

[uc1, uprimc1]=utilramsey(c1);
[uc2, uprimc2]=utilramsey(c2);
[gk2, gprimk2]=transramsey(k2, c2);

y=discount*gprimk2.*uprimc2-uprimc1;
```

_____ Koniec pliku *eqeulerramsey.m* _____

6.4 Numeryczne rozwiązanie modelu Ramseya

Przed przystąpieniem do podania funkcji zwracającej resztę równania Eulera w węzłach kolokacji warto najpierw określić poziom kapitału i konsumpcji w stanie ustalonym, by wiedzieć na jakim przedziale aproksymować funkcję polityki (przedział ten powinien zawierać stan ustalony). Stan ustalony jest określony przez układ dwóch równań:

$$\beta[1 - \delta + f'(k^*)]u'(c^*) - u'(c^*) = 0$$

oraz

$$k^* = g(k^*, c^*).$$

Funkcja `ramseystst` oblicza resztę obu równań korzystając z funkcji `eqeulerramsey` i `transramsey`. `x(1)` odpowiada poziomowi kapitału, `x(2)` – konsumpcji.⁴

_____ Plik `ramseystst.m` _____

```
function y=ramseystst(x)
y=zeros(2,1);
y(1)=eqeulerramsey(x(1), x(2), x(1), x(2));
y(2)=transramsey(x(1), x(2))-x(1);
```

_____ Koniec pliku `ramseystst.m` _____

Korzystając z funkcji `newton` możemy rozwiązać układ równań określający stan ustalony.

```
octave:1> newton ('ramseystst', [1 1]')
ans =
```

```
7.1278
1.5681
```

Wiedząc jaki jest stan ustalony możemy przystąpić do napisania funkcji `resideulerramsey` podającej resztę równania Eulera w węzłach kolokacji (lewe strony układu równań (6.6)). Przyjmujemy, że polityka jest aproksymowana dla poziomów kapitału z przedziału $[4, 11]$ przy użyciu wielomianu 9-go stopnia. Do równania Eulera za dzisiejszy kapitał podstawiamy wektor węzłów Czebyszewa, zaś dzisiejszą konsumpcję obliczamy mnożąc macierz Φ ($\Phi_{i,j} = T_{j-1}(x_i)$) przez wektor współczynników szeregu Czebyszewa. Jutrzejszy kapitał wyznaczamy korzystając z funkcji `transramsey`, zaś do obliczenia jutrzejszej konsumpcji używamy funkcji `chebval` pozwalającej obliczyć wartość szeregu Czebyszewa aproksymującego $c(k)$ dla dowolnego k .

_____ Plik `resideulerramsey.m` _____

```
function ret=resideulerramsey(v)
kmin=4;
kmax=11;
```

⁴ Oczywiście stan ustalony modelu Ramseya dla przyjętych postaci funkcji produkcji i użyteczności można wyznaczyć analitycznie, odpowiednio upraszczając powyższe równania. Procedura stosowana przez nas w tym miejscu jest ogólna i znajduje zastosowanie w przypadku, gdy analityczne znalezienie stanu ustalonego jest niemożliwe lub bardzo kosztowne.

6 Metoda kolokacji

```

k1=chebnod(10, kmin, kmax);
c1=chebphi(10)*v;
k2=transramsey(k1, c1);
c2=chebval(v, k2, kmin, kmax);

ret=eqeulerramsey(k1, c1, k2, c2);

```

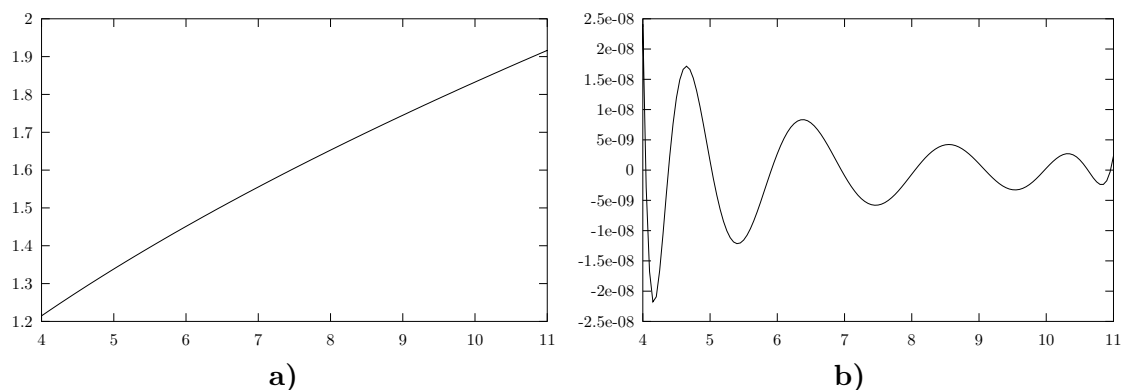
Koniec pliku *resideulerramsey.m*

Możemy teraz wykorzystać funkcję `newton` do wyznaczenia współczynników naszej aproksymacji. W startowym przybliżeniu przyjmujemy, że konsumpcja jest równa 1,5 dla $k = 7,5^5$ i jest rosnącą funkcją poziomu kapitału.

```
octave:2> a=newton ('resideulerramsey', [1.5 1 0 0 0 0 0 0 0 0]');
```

Możemy sprawdzić jak wygląda nasza funkcja polityki (rys. 6.3a).

```
octave:3> x=(4:.05:11)'; y=chebval (a, x, 4, 11); plot(x,y)
```



Rysunek 6.3: Model Ramseya: **a)** optymalna polityka $c(k)$, **b)** reszta równania Eulera

Poniższe komendy pozwalają sprawdzić zachowanie się błędu równania Eulera na całym przedziale. Jak widać z rys. 6.3b uzyskane przez nas przybliżenie jest całkiem dobre.

```

octave:4> z=transramsey (x,y);
octave:5> R=eqeulerramsey (x, y, z, chebval (a, z, 4, 11));
octave:6> plot(x, R)

```

Oczywiście bardziej nawet niż sama funkcja polityki interesuje nas ewolucja modelowanej gospodarki. Poniższe komendy pozwalają zasymulować ewolucję kapitału w czasie zaprezentowaną na rysunku 6.4a. Przyjeliśmy początkowy poziom kapitału równy 5. Symulacja trwa 100 okresów.

```

octave:7> K=zeros(100,1); K(1)=5;
octave:8> for i=(2:100)
> K(i)=transramsey (K(i-1), chebval (a, K(i-1), 4, 11));
> end

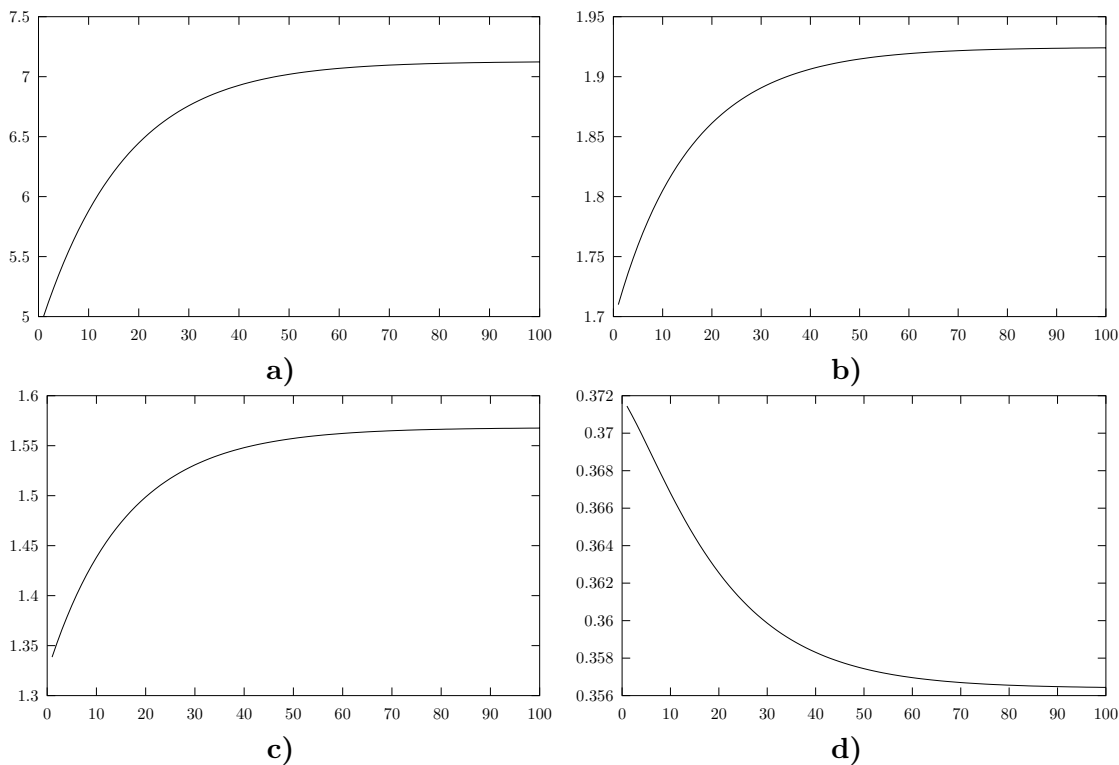
```

⁵ Przy liniowej transformacji przedziału $[4, 11]$ na przedział $[-1, 1]$ $7,5$ przechodzi na 0 .

Zadania

Mając ewolucję kapitału, korzystając z funkcji produkcji, aproksymacji funkcji polityki $c(k)$ i prostej tożsamości rachunkowej możemy zobaczyć jak zmieniają się w czasie produkt, konsumpcja i inwestycje (rys. 6.4b-d).

```
octave:9> Y=prodframsey (K);
octave:10> C=chebval (a, K, 4, 11);
octave:11> I=Y-C;
```



Rysunek 6.4: Model Ramseya: ewolucja **a)** kapitału, **b)** produkcji, **c)** konsumpcji i **d)** inwestycji

Zadania

Zadanie 6.1 Rozwiąż numerycznie model Ramseya z endogeniczną podażą pracy. Czas wolny wchodzi do funkcji użyteczności chwilowej w następujący sposób ($l \in [0, 1]$ oznacza podaż pracy):

$$u(c, l) = \frac{(c^\tau (1-l)^{1-\tau})^{1-\theta}}{1-\theta}.$$

Funkcja produkcji to funkcja Cobba-Douglasa:

$$f(k, l) = k^\alpha l^{1-\alpha}.$$

Problem ma standardową postać:

$$\max \sum_{t=0}^{\infty} \beta^t u(c_t, l_t) \quad \text{pw.} \quad c_t + i_t \leq f(k_t, l_t), \quad k_{t+1} = (1-\delta)k_t + i_t.$$

6 Metoda kolokacji

1. Zapisz równanie Bellmana.
2. Wprowadź warunki pierwszego rzędu.
3. Zmień odpowiednio funkcje `prodframsey`, `transramsey` i `utilramsey`. Napisz funkcję obliczającą resztę równania Eulera i rozwiąż numerycznie równanie kolokacji. Sprawdź zachowanie się reszt. Przyjmij $\tau = 0,35$, zaś pozostałe parametry takie, jak w treści rozdziału.
4. Dokonaj symulacji kapitału, podaży pracy, produkcji i konsumpcji.

Wskazówka: Optymalne sterowania c i l są ze sobą związane warunkiem pierwszego rzędu, więc wystarczy aproksymować tylko jedną funkcję polityki c lub l (jeśli mamy jedną, drugą możemy znaleźć z warunku pierwszego rzędu, który w tym przypadku ma bardzo prostą postać.)

Dodatki

Dodatek A

Podstawy obsługi programów MATLAB i OCTAVE

Poniższa prezentacja programów MATLAB i OCTAVE jest siłą rzeczy bardzo skrótowa i ograniczona do niezbędnego dla samowystarczalności książki minimum. Więcej informacji na temat działania programów MATLAB i OCTAVE można znaleźć w dołączonej do nich dokumentacji lub na stronach <http://www.mathworks.com> i <http://www.octave.org>.

A.1 Operacje arytmetyczne. Zmienne

Najprostszym zastosowaniem programów MATLAB/OCTAVE są obliczenia arytmetyczne na liczbach rzeczywistych i zespolonych. Wystarczy zapisać nasze działanie i nacisnąć <ENTER>. Popatrzmy:

```
octave:1> 3/4-1
ans = -0.25000
octave:2> (1+i)/(1-i)
ans = 0 + 1i
```

W drugim przykładzie i oznacza jednostkę urojoną czyli $\sqrt{-1}$. MATLAB/OCTAVE mają zapisane podstawowe stałe matematyczne π i e . Przy okazji zwróćmy uwagę, że MATLAB/OCTAVE korzystają z angielskiego sposobu zapisu liczb dziesiętnych z kropką jako separatorem.

```
octave:3> pi
pi = 3.1416
octave:4> e
e = 2.7183
```

Oprócz podstawowych działań arytmetycznych użytkownik ma dostęp do praktycznie wszystkich najważniejszych funkcji (pamiętajmy o angielskim nazewnictwie!), w tym funkcji trygonometrycznych (`sin`, `cos`, `tan`, `asin`, `acos`, `atan`), logarytmu (`log`) i eksponenty (`exp`).

A.2 Wektory i macierze

```
octave:5> cos(pi)
ans = -1
octave:6> log(e)
ans = 1
```

Bardzo ważną operacją jest potęgowanie i pierwiastkowanie. Do potęgowania służy operator „^”, pierwiastek kwadratowy wyciągamy przy użyciu funkcji `sqrt` (ang. *square root*).

```
octave:7> 2^10
ans = 1024
octave:8> sqrt(1024)
ans = 32
```

Pokażemy teraz jak korzystać ze zmiennych. Najprostsze przypisanie (i jednoczesna deklaracja zmiennej) zaprezentowane jest poniżej.

```
octave:9> a=3
a = 3
```

Zauważmy, że OCTAVE pokazało wynik operacji (przypisanie zmiennej `a` liczby 3). Jeśli, jak poniżej, po przypisaniu postawimy średnik „;” wynik operacji nie będzie pokazany.

```
octave:10> b=4;
```

W poniższym przykładzie obliczamy średnią z liczb `a` i `b` i zapisujemy wynik w zmiennej `c`.

```
octave:11> c=(a+b)/2
c = 3.5000
```

Ćwiczenie A.1 Zapisz w zmiennej `d` liczbę $\pi/2$, następnie dokonaj przypisania `d=sin(d)/2`. Jaka będzie nowa wartość `d`?

A.2 Wektory i macierze

Najważniejszą cechą programów MATLAB i OCTAVE jest sprawne operowanie wektorami i macierzami odbywające się dzięki odwołaniom do bibliotek BLAS (ang. *Basic Linear Algebra Subroutines*) i LAPACK (ang. *Linear Algebra Package*). Pokażemy teraz jak deklarować wektory i macierze, odwoływać się do nich i wykonywać najprostsze operacje.

A.2.1 Deklaracje i przypisania wektorów i macierzy

Wektory, podobnie jak zmienne, deklarujemy przez przypisanie. Współrzędne wektora podajemy w nawiasach kwadratowych „[]” oddzielając je od siebie przecinkiem lub spacją.¹ Oto przykład:

```
octave:12> x=[-1, 0, 3]
x =
```

```
-1  0  3
```

Do współrzędnych odwołujemy się podając ich numer w nawiasach zwykłych „()”:

```
octave:13> x(1)
ans = -1
octave:14> x(2)
ans = 0
octave:15> x(3)
ans = 3
```

Ćwiczenie A.2 Jaki będzie wynik przypisania $x(2)=-4$?

Deklaracja i przypisanie macierzy wygląda podobnie jak w przypadku wektorów. Kolejne kolumny w wierszu oddzielamy spacją lub przecinkiem, zaś wiersze średnikiem „;”. Poniżej pokazujemy inicjalizację macierzy 2×2 .

```
octave:16> A=[-3, 4; 1, 2]
A =
```

```
-3  4
 1  2
```

Odwołanie do konkretnej pozycji macierzy także przypomina odwołania do wektorów. W nawiasie zwykłym podajemy najpierw numer wiersza, potem kolumny:

```
octave:17> A(1,2)
ans = 4
```

Często przydaje się odwołanie się do całego wiersza/kolumny macierzy. W takich sytuacjach podajemy numer wiersza (kolumny), zaś w miejsce numeru kolumny (wiersza) wpisujemy dwukropek „:”.

```
octave:18> A(1,:)
ans =
```

```
-3  4
```

```
octave:19> A(:,2)
ans =
```

```
4
 2
```

¹ Należy uważać przy oddzielaniu wyrażeń spacjami ze względu na możliwość występowania dwuznaczności. Np. $[1 \ -2 \ 3]$ to wektor $(1, -2, 3)$, zaś $[1 \ - \ 2 \ 3]$ to wektor $(-1, 3)$.

A.2.2 Wektory i macierze specjalne

Wpisywanie za każdym razem ręcznie macierzy (np. przy deklaracji) jest uciążliwe i w wielu przypadkach niewykonalne.² Najczęściej macierze i wektory inicjalizuje się wykorzystując pewne funkcje, których wynikiem są macierze jakich oczekujemy. Poniżej podajemy kilka najważniejszych funkcji generujących macierze i wektory.

Macierz jednostkową $n \times n$ tworzymy używając komendy `eye` (od brzmienia angielskiego *I* – standardowego oznaczenia macierzy jednostkowej).

```
octave:20> eye(3)
```

```
ans =
```

```
1 0 0
0 1 0
0 0 1
```

Macierze $n \times m$ składające się z samych zer lub jedynek tworzymy używając funkcji `zeros` i `ones` podając liczbę wierszy i kolumn. W przypadku podania jednej liczby funkcje te tworzą macierze kwadratowe o zadanym wymiarze.

```
octave:21> zeros(3,2)
```

```
ans =
```

```
0 0
0 0
0 0
```

```
octave:22> ones(2,4)
```

```
ans =
```

```
1 1 1 1
1 1 1 1
```

Często wykorzystuje się (np. w pętlach) wektory których kolejne wyrazy są elementami ciągu arytmetycznego. Taki wektor tworzy się podając w nawiasie zwykłym pierwszą i ostatnią liczbę ciągu oddzielone dwukropkiem.

```
octave:23> (1:6)
```

```
ans =
```

```
1 2 3 4 5 6
```

Jeśli zależy nam na utworzeniu ciągu arytmetycznego o innej różnicy należy ją podać pomiędzy pierwszą i ostatnią liczbą ciągu oddzielając dwukropkiem.

```
octave:24> (-2:0.5:2)
```

```
ans =
```

```
-2.0000 -1.5000 -1.0000 -0.5000 0.0000 0.5000 1.0000 1.5000
2.0000
```

² Jak wyglądałoby wpisanie macierzy jednostkowej 100×100 ?

Ostatnią bardzo użyteczną klasą macierzy (i wektorów) są macierze wypełnione liczbami pseudolosowymi. Funkcja `rand` służy do wypełniania liczbami pseudolosowymi z rozkładu jednostajnego na $(0, 1)$, zaś funkcja `randn` liczbami ze standardowego rozkładu normalnego $N(0, 1)$.

```
octave:25> rand(2)
```

```
ans =
```

```
0.011155 0.193487
0.053544 0.545317
```

```
octave:26> rand(2,3)
```

```
ans =
```

```
0.98109 0.10993 0.92111
0.70334 0.29451 0.78699
```

```
octave:27> randn(2,3)
```

```
ans =
```

```
-0.60000 1.22904 -0.87689
-0.42401 0.49441 -0.85250
```

A.2.3 Działania na wektorach i macierzach

Mnożenie przez skalar

Operacje mnożenia wektorów i macierzy przez skalary przeprowadza się naturalnie używając zwykłego operatora mnożenia „*”. Poniżej prezentujemy dwa przykłady.

```
octave:28> x=[1 -4 3 2.5]
```

```
x =
```

```
1.0000 -4.0000 3.0000 2.5000
```

```
octave:29> 2*x
```

```
ans =
```

```
2 -8 6 5
```

```
octave:30> 3*eye(2)
```

```
ans =
```

```
3 0
0 3
```

Dodawanie

Macierze i wektory dodajemy naturalnie używając znaku „+”. W przypadku niezgodności wymiarów wektorów (macierzy) operacja zakończy się błędem.

A.2 Wektory i macierze

```
octave:31> [1 2 3]+[-4 -3 -2]
ans =
```

```
-3 -1 1
```

```
octave:32> [1 2 3]+[-4 -3 -2 0]
error: operator +: nonconformant arguments (op1 is 1x3, op2 is 1x4)
error: evaluating binary operator '+' near line 30, column 8
octave:32> eye(2)+ones(2)
ans =
```

```
2 1
1 2
```

Mnożenie

Macierze $n \times m$ i $m \times k$ mnożymy używając zwykłego operatora mnożenia „*”. W przypadku niezgodności wymiarów operacja zakończy się błędem.

```
octave:33> A=[1 3; 2 5]
A =
```

```
1 3
2 5
```

```
octave:34> B=[1 4 5; 3 4 3]
B =
```

```
1 4 5
3 4 3
```

```
octave:35> A*B
ans =
```

```
10 16 14
17 28 25
```

```
octave:36> B*A
error: operator *: nonconformant arguments (op1 is 2x3, op2 is 2x2)
error: evaluating binary operator '*' near line 34, column 2
```

Mnożenia wektora przez macierz dokonuje się podobnie. W poniższym przykładzie próba pomnożenia macierzy A przez wektor wierszowy (macierz 1×2) zakończy się błędem.

```
octave:36> x=[-3 4]
x =
```

```
-3 4
```



```
octave:37> A*x
error: operator *: nonconformant arguments (op1 is 2x2, op2 is 1x2)
error: evaluating binary operator '*' near line 35, column 2
```

Możemy dokonać transpozycji wektora x używając pojedynczego znaku cudzo-
słowa „'”. Wtedy operacja mnożenia macierzy A i wektora x stanie się możliwa.

```
octave:37> A*x'
ans =

     9
    14
```

Działania „wyraz po wyrazie”

Czasami zachodzi potrzeba wykonania działań na wektorach lub macierzach wy-
raz po wyrazie. Np. wyobraźmy sobie, że chcemy podnieść każdą liczbę w pewnym
wektorze do kwadratu (przemnożyć przez nią samą). Użycie operatora mnożenia
zakończy się w takiej sytuacji błędem:

```
octave:38> [1 2 3]*[1 2 3]
error: operator *: nonconformant arguments (op1 is 1x3, op2 is 1x3)
error: evaluating binary operator '*' near line 36, column 8
```

Aby programy MATLAB lub OCTAVE wiedziały, że naszym celem jest działanie
wyraz po wyrazie, musimy przed znakiem działania dać kropkę. W naszym przy-
padku używamy operatora „.*”.

```
octave:38> [1 2 3].*[1 2 3]
ans =

     1     4     9
```

Oczywiście, równie dobrze zamiast mnożyć wektor przez siebie można podnieść
go do kwadratu. Znowu musimy jednak pamiętać o kropce.

```
octave:39> [1 2 3].^2
ans =

     1     4     9
```

Funkcje arytmetyczne standardowo działają wyraz po wyrazie. Oto przykład.

```
octave:40> sin([-pi -pi/2 0 pi/2 pi])
ans =

-0.00000 -1.00000  0.00000  1.00000  0.00000
```

Warto wiedzieć, że do wektora (i macierzy) można dodać skalar i operacja taka
nie zakończy się błędem – skalar zostanie dodany do każdego z wyrazów.

A.3 Programowanie w języku MATLAB/OCTAVE

```
octave:41> zeros(1,5)-2
ans =
```

```
-2 -2 -2 -2 -2
```

Jeśli na przykład chcemy utworzyć wektor (nazwijmy go z) mający na pierwszych 9-ci u pozycjach π zaś na ostatniej 10-tej e wystarczy wykonać następujące operacje:

```
octave:42> z=zeros(1,10)+pi
z =
```

```
3.1416 3.1416 3.1416 3.1416 3.1416 3.1416 3.1416 3.1416 3.1416
3.1416
```

```
octave:43> z(10)=e
z =
```

```
3.1416 3.1416 3.1416 3.1416 3.1416 3.1416 3.1416 3.1416 3.1416
2.7183
```

A.3 Programowanie w języku Matlab/Octave

Programy MATLAB i OCTAVE są czymś więcej niż tylko zaawansowanymi kalkulatorami przeznaczonymi do operacji na macierzach, są także interpreterami bardzo prostego i elastycznego języka programowania przeznaczonego do zagadnień numerycznych. Poniżej pokażemy jak deklarować i zapisywać proste funkcje w języku MATLAB/OCTAVE oraz przybliżymy podstawowe konstrukcje tego języka.

A.3.1 Skrypty i funkcje

MATLAB i OCTAVE pozwalają zapisywać zestawy swoich poleceń w plikach z rozszerzeniem `.m`. Taki plik tekstowy zawierający listę poleceń MATLAB/OCTAVE nazywamy skrypcem. Aby wykonać zestaw poleceń w skrypcie należy po prostu wpisać jego nazwę (bez rozszerzenia `.m`) i nacisnąć `<ENTER>`. Aby plik został wykonany, musi znajdować się w jednym z katalogów wskazywanych przez zmienną `path` (w MATLABie) lub zmienną `LOADPATH` (w OCTAVE). Jeśli np. używamy OCTAVE pod Linuksem i chcemy zapisywać swoje skrypty i programy w podkatalogu `mfiles/` katalogu domowego (np. `/home/gklima/mfiles/`) najlepiej wykonać na początku każdej sesji poniższą komendę.

```
octave:1> LOADPATH = [DEFAULT_LOADPATH ':' '~/mfiles//']
```

Jeśli używamy MATLABa pod MS Windows i zapisujemy swoje pliki `.m` w katalogu `c:\mfiles` należy wykonać na starcie poniższą komendę.³

³ Powyższe operacje można zautomatyzować. W przypadku korzystania z OCTAVE wystarczy utworzyć plik `.octaverc` w katalogu domowym i wpisać tam komendę ustawiającą `LOADPATH` (najlepiej dodając na koniec średnik). Jeśli używamy MATLABa musimy znaleźć w strukturze katalogów instalacji plik `matlabrc` i tam dopisać komendę odpowiednio ustawiającą ścieżkę.

```
>> path(path, 'c:\mfiles')
```

Należy pamiętać, że pliki funkcji i skryptów są plikami *tekstowymi*, a więc nie należy ich tworzyć w żadnym procesorze tekstu, np. MS Word czy Open Office Writer, tylko w zwykłym edytorze.

Najczęściej skrypty przydają się do zautomatyzowanej inicjalizacji zmiennych. Oto przykład bardzo prostego skryptu, który inicjalizuje zmienne `alpha`, `beta` i `delta`. Nasz plik nosi nazwę `example0.m`.

```
_____ Plik example0.m _____
alpha=1/3;
beta=1/1.04;
delta=0.0198;
_____ Koniec pliku example0.m _____
```

Poniższe komendy wykonują nasz skrypt i wykorzystują zmienną `alpha` do prostych obliczeń.

```
octave:1> example0
octave:2> 100^alpha
ans = 4.6416
```

Ćwiczenie A.3 Napisz skrypt inicjalizujący macierz A 2×10 , której pierwszy wiersz to liczby od 0 do 9 zaś drugi to liczby od 9 do 0. Nie wpisuj elementów tej macierzy ręcznie, tylko użyj polecenia `zeros` i skorzystaj z operatora „:”.

Funkcje są zestawami poleceń, które działają na pewnych podanych przez użytkownika argumentach i zwracają wynik. Funkcje tworzone przez użytkownika są, podobnie jak skrypty, zapisywane w plikach z rozszerzeniem `.m`, które należy umieścić w odpowiednim katalogu. Do tej pory poznaliśmy kilka standardowych funkcji MATLAB/OCTAVE, np. funkcję `sin` czy `eye`. Część standardowych funkcji jest skompilowana z MATLABem czy OCTAVE, część jest dostarczona w postaci plików z rozszerzeniem `.m`, napisanych w ten sam sposób, w jaki piszemy funkcje poniżej.⁴

Aby interpreter poprawnie zrozumiał i wykonał oczekiwane przez nas operacje musimy podać mu kilka podstawowych informacji. Po pierwsze, deklaracja funkcji musi zaczynać się od słowa kluczowego `function`. Po drugie, każda funkcja musi mieć unikalną nazwę, która musi się zgadzać z nazwą pliku, w którym funkcję zapisujemy. W większości języków programowania wynik funkcji jest zwracany przez odwołanie do specjalnego słowa kluczowego, w MATLABie i OCTAVE tak nie jest – zwrócenie wyniku odbywa się dzięki przypisaniu go do pewnej *lokalnej* (czyli rozumianej tylko wewnątrz funkcji) zmiennej, zadeklarowanej w preambule funkcji. Argumenty, na których funkcja działa muszą też mieć swoje lokalne nazwy zadeklarowane w preambule. W odróżnieniu od większości języków programowania nie ma konieczności deklarowania typu (np. macierz, wektor, liczba) ani argumentu, ani wyniku. Jeśli typ argumentu podanego przez użytkownika nie pozwala na wykonanie operacji zakodowanych w funkcji jej wykonanie zakończy się błędem.

Oto schemat pliku `.m` zawierającego funkcję:

⁴ Kod funkcji zawartej w pliku `.m`, lub informację, że jest to funkcja skompilowana można uzyskać korzystając z komendy `type`, po której (po spacji) należy podać nazwę interesującej nas funkcji.

A.3 Programowanie w języku MATLAB/OCTAVE

```
function lokalna nazwa wyniku = nazwa funkcji (lokalna nazwa argumentu)
...
komendy
...
lokalna nazwa wyniku = ...
```

Poniższa przykładowa funkcja `example1` zwraca iloczyn argumentu przez jego transpozycję.

```
----- Plik example1.m -----
function y=example1(x)
y=x*x';
----- Koniec pliku example1.m -----
```

Oto przykłady jej działania:

```
octave:1> example1 ([1 2 3])
ans = 14
octave:2> example1 ([1 2 3]')
ans =
```

```
1 2 3
2 4 6
3 6 9
```

```
octave:3> example1 (ones(2,3))
ans =
```

```
3 3
3 3
```

Należy pamiętać, że zmienne `x` i `y` są zmiennymi lokalnymi dla funkcji `example1` i następujące przypisania w obszarze roboczym nie będą miały wpływu na jej działanie, ani też jej wykonanie nie będzie miało wpływu na zawartość tychże zmiennych.

```
octave:4> x=5; y=3;
octave:5> example1 ([1 2 3])
ans = 14
octave:6> x
x = 5
octave:7> y
y = 3
```

Ćwiczenie A.4 Napisz funkcję, która dla podanej macierzy A obliczy $(A+A')/2$. Zauważ, że działanie to ma sens tylko gdy A jest kwadratowa. Spróbuj wykonać tę funkcję na jakiejś macierzy niekwadratowej lub wektorze. Co się stanie?

Funkcje w MATLABie i OCTAVE mogą, podobnie jak we wszystkich standardowych językach programowania, brać więcej niż jeden argument. Jedną z ciekawych cech języka MATLAB/OCTAVE, jest to, że funkcje mogą zwracać więcej niż jeden wynik. Ogólny schemat funkcji biorącej n argumentów i zwracającej m wyników podajemy poniżej.

```
function [wynik1,...,wynikm] = nazwa_funkcji (arg1, ..., argn)
...
komendy
...
wynik1 = ...
...
wynikm = ...
```

Poniższa funkcja jest niejako rozszerzeniem funkcji `example1`. Jako drugi wynik zwraca iloczyn transpozycji argumentu przez samego siebie.

```
----- Plik example2.m -----
function [y, yy]=example2(x)

y=x*x';
yy=x'*x;
----- Koniec pliku example2.m -----
```

Przy standardowym wywołaniu podawany jest tylko pierwszy wynik.

```
octave:1> example2([1 2 3])
ans = 14
octave:2> a=example2([1 2 3])
a = 14
```

Jeśli chcemy na raz przypisać oba wyniki do dwóch zmiennych wystarczy zrobić jak poniżej.

```
octave:3> [a,b]=example2([1 2 3])
a = 14
b =

    1    2    3
    2    4    6
    3    6    9
```

Ćwiczenie A.5 Napisz funkcję `sredwar`, która dla danego wektora wierszowego pewnych obserwacji oblicza średnią z próby i wariancję. **Wskazówka:** do określenia wielkości próby użyj funkcji `size`. Funkcja ta zwraca wektor 1×2 (najpierw liczbę wierszy, potem kolumn), więc najlepiej użyć przypisania `n=size(x,2)` (patrz `help size`). Do zsumowania liczb w wektorze użyj funkcji `sum`.

Ćwiczenie A.6 Napisz funkcję `korelacja`, która dla dwóch wektorów wierszowych oblicza ich korelację. Do obliczenia wartości średniej i wariancji obu wektorów użyj napisanej przez siebie wcześniej funkcji `sredwar`.

A.3.2 Konstrukcje warunkowe i pętle

Warunkowe wykonywanie poleceń oraz pętle są najważniejszymi konstrukcjami każdego języka programowania – pozwalają programom podejmować decyzje i automatyzować powtarzanie operacji.

A.3 Programowanie w języku MATLAB/OCTAVE

if, else i elseif

Najprostsza konstrukcja warunkowa sprowadza się do uzależnienia wykonania polecenia (serii poleceń), od spełnienia pewnego warunku. Schemat takiej konstrukcji przedstawiamy poniżej.

```
if warunek
    ...
    komendy, które mają być wykonane jeśli warunek jest spełniony
    ...
end
```

Najprostsze warunki mają postać relacji, np. $a > b$, $a < b$, $a == b$ (a równe b), $a <= b$ (mniejsze lub równe), $a >= b$, $a \neq b$ (różne). Bardziej złożoną postać warunku można uzyskać korzystając z operatorów logicznych & (i), | (lub) i ~ (nie). Np. konstrukcja $(a >= 0) | (b >= 0)$ oznacza „przynajmniej jedna z liczb a, b jest nieujemna”, zaś $(a < 0) \& (b > 1)$ – „a jest mniejsze od zera i b większe od 1”.

Najczęściej chcemy wskazać działania, które mają zostać wykonane przy spełnieniu warunku i alternatywne działania, które mają być wykonane w przeciwnym przypadku. Do konstrukcji tego typu używamy słowa kluczowego **else**.

```
if warunek
    ...
    komendy, które mają być wykonane jeśli warunek jest spełniony
    ...
else
    ...
    komendy, które mają być wykonane jeśli warunek nie jest spełniony
    ...
end
```

Poniższa funkcja losuje liczbę (pseudolosową) ze zbioru $\{0, 1\}$ z prawdopodobieństwem p sukcesu (jedyński) korzystając z generatora liczb pseudolosowych o rozkładzie równomiernym na $[0, 1]$ wbudowanego w funkcję **rand**.

Plik *example3.m*

```
function ret=example3(p)

if rand(1,1)<p
    ret=1;
else
    ret=0;
end
```

Koniec pliku *example3.m*

Ćwiczenie A.7 Uzupełnij funkcję **korelacja** z ćwiczenia A.6 o kontrolę zgodności wymiarów obu wektorów danych. Gdy wymiary się nie zgadzają funkcja powinna obliczać korelację pierwszych n wyrazów, gdzie n to liczba danych w wektorze z mniejszą liczbą obserwacji. **Wskazówka:** do wybrania pierwszych n wyrazów wektora x o długości większej niż n wykorzystaj operator $:$ pisząc $x(1:n)$.

Czasami występują sytuacje, w których chcemy w razie niespełnienia jednego warunku sprawdzić jakiś inny i wykonać odpowiednie działanie według schematu: „jeśli spełnione a to zrób to, jeśli nie a i spełnione b , to zrób coś innego. Do takich konstrukcji używa się słowa kluczowego `elseif`. Ogólna konstrukcja z użyciem `elseif` pokazana jest poniżej.

```
if warunek1
    ...
    komendy, które mają być wykonane jeśli warunek1 jest spełniony
    ...
elseif warunek2
    ...
    komendy, które mają być wykonane jeśli warunek1 nie jest, a warunek2 jest
    spełniony
    ...
end
```

Konstrukcje warunkowe można zagnieżdżać jedna w drugiej.

Pętle while

Często – np. w iteracyjnych algorytmach numerycznych – pojawia się potrzeba powtarzanie pewnej operacji tak długo jak długo spełniony jest pewien warunek, lub też do momentu spełnienia pewnego warunku. Pętle tego typu tworzy się używając słowa kluczowego `while` według prezentowanego poniżej schematu.

```
while warunek
    ...
    komendy, które mają być wykonywane jak długo warunek jest spełniony
    ...
end
```

Poniższa funkcja oblicza pierwiastek równania $\cos x = x$ z zadaną dokładnością używając tzw. metody bisekcji. Szukanie pierwiastka naszego równania jest równoważne z szukaniem miejsca zerowego funkcji $f(x) = \cos x - x$. Wiemy, że $f(0) > 0$ i jednocześnie $f(\frac{\pi}{2}) < 0$. Z ciągłości f wynika, że pierwiastek leży gdzieś między 0 i $\frac{\pi}{2}$. Załóżmy, że mamy dwie liczby a_n i b_n , o których wiemy, że $f(a_n) > 0$ i $f(b_n) < 0$. Obliczamy $y = f(\frac{a_n+b_n}{2})$ (wartość funkcji w środku przedziału). Jeśli $y > 0$ to znaczy, że pierwiastek, leży między $\frac{a_n+b_n}{2}$ i b_n , więc przyjmujemy $a_{n+1} = \frac{a_n+b_n}{2}$ i $b_{n+1} = b_n$, gdy $y < 0$ przypadku bierzemy $a_{n+1} = a_n$ i $b_{n+1} = \frac{a_n+b_n}{2}$. Jeśli $y = 0$, to znaczy, że znaleźliśmy dokładnie szukany pierwiastek. W każdej iteracji przedział zawierający szukany pierwiastek (przedział (a_n, b_n) zmniejsza się o połowę. Jeśli $b_n - a_n < \varepsilon$ i ε jest satysfakcjonującą nas dokładnością procedurę kończymy.

----- Plik `example4.m` -----

```
function ret=example4(tol)
a=0;
```

A.3 Programowanie w języku MATLAB/OCTAVE

```

b=pi/2;
y=1;

while ((b-a)>tol)&(y~=0)
    c=(a+b)/2;
    y=cos(c)-c;
    if y>0
        a=c;
    else
        b=c;
    end
end

ret=b;

```

Koniec pliku *example4.m*

Używając pętli `while` należy sprawdzać, czy nie zakodowaliśmy pętli nieskończonej, czyli takiej, w której warunek będzie zawsze spełniony. Jeśli podejrzewamy, że warunek może być zawsze spełniony lub obliczenia w jednej pętli trwają długo, a nasz iteracyjny schemat może zapewnić wyjście z pętli dopiero po bardzo wielu iteracjach, najlepiej wprowadzić pewną zmienną zliczającą iteracje (np. n) i uzupełnić warunek wyjściowy dodatkowym warunkiem $n < N$, gdzie np. $N=1000$. Przykładowy schemat takiego zabezpieczenia prezentujemy poniżej.

```

N=1000; n=0;
while ( warunek )&(n<N)
    n=n+1;
    ...
    komendy
    ...
end

```

Pętle for

Pętli `for` używamy, gdy chcemy wykonać z góry zadaną liczbę iteracji, z których każda jest indeksowana elementami pewnego wektora lub macierzy. W kolejnej iteracji zmienna indeksująca przyjmuje kolejną wartość z zadanego wektora (lub macierzy). Oto ogólny schemat pętli `for`.

```

for zmienna indeksująca = wektor
    ...
    komendy
    ...
end

```

Poniższa funkcja wyznacza pierwszych n wyrazów ciągu zadanego rekurencyjnie $a_n = a_{n-1} + a_{n-2}$ z warunkiem początkowym $a_1 = a_2 = 1$.

Plik *example5.m*

```
function ret=example5(n)
x=ones(1,n);
if n>2
    for i=(3:n)
        x(i)=x(i-1)+x(i-2);
    end
end
ret=x;
```

Koniec pliku *example5.m*

A.3.3 Parę uwag o dobrym stylu programowania

Wektoryzacja

Język programowania MATLAB/OCTAVE jest językiem interpretowanym i pętle działają w nim bardzo wolno. Najważniejsze dla zapewnienia wydajności programów jest maksymalne wykorzystanie wkompilowanych bibliotek BLAS i LAPACK bardzo efektywnie operujących na wektorach i macierzach. Wszędzie, gdzie operując na macierzach i wektorach można wykorzystywać takie funkcje jak `sum`, operator `:`, czy działania wyraz po wyrazie, *nie należy* stosować pętli `for`.

W wielu przypadkach niewielki wysiłek programistyczny pozwala zapewnić by nasza funkcja przyjmowała jako argumenty nie tylko liczby, lecz także wektory wykonując na każdej liczbie tę samą operację, lub zamiast wektora przyjmowała macierz i wykonywała na każdej kolumnie (lub wierszu) tę samą operację.

Przejrzystość kodu

Najważniejszym zaleceniem przy pisaniu w każdym języku programowania jest jasne prezentowanie hierarchizacji kodu poprzez stosowanie wcięć, tym większych im niżej w hierarchii kodu jest dana komenda. Taki sposób pisania kodu ułatwia szybkie rozpoznanie, gdzie rozpoczyna i kończy się pętla czy konstrukcja warunkowa. Warto też oddzielać pustymi liniami kolejne części algorytmu, np. inicjalizację pomocniczych zmiennych od głównej pętli.

Komentarze

Pisząc większe funkcje lub zestawy funkcji warto w formie komentarza dodać kilka uwag, o tym, co dana funkcja robi, jakie przyjmuje argumenty, jaki zastosowano algorytm itp. Komentarze rozpoczynają się od znaku `%` W środowisku użytkowników MATLABa i OCTAVE przyjęło się umieszczać wszelkie informacje przed lub tuż pod deklaracją funkcji, przed jej kodem.

Ćwiczenie A.8 Znajdź w strukturze katalogów swojej instalacji MATLABa (np. `c:\matlab`) lub OCTAVE (np. `/usr/share/octave`) pliki `.m` i przejrzyj ich strukturę.

A.4 Kilka dodatkowych możliwości programów Matlab i Octave

A.4.1 Wczytywanie i zapisywanie danych z/do plików

Zdarza się, że dane, na których chcemy pracować zapisane są w pewnym pliku. Do ich wczytywania służy funkcja `load`. `load` obsługuje kilka formatów zapisu danych,⁵ my ograniczymy się do przypadku, gdy dane są zapisane w formacie tekstowym.

Założmy, że w katalogu, z którego uruchomiliśmy MATLABa lub OCTAVE (ew. w którymś z katalogów w ścieżce dostępu) znajduje się plik `data0` o następującej zawartości:

```

_____ Plik data0 _____
1 4
54 4
45 68
345 8
2 -1
_____ Koniec pliku data0 _____

```

Aby wczytać zawartość tego pliku do zmiennej o takiej samej nazwie jak nazwa pliku (w naszym przykładzie `data0`) wystarczy napisać „`load nazwapliku`” i możemy już normalnie operować na naszej zmiennej.

```

octave:1> load data0
octave:2> data0'
ans =

```

```

     1     54     45    345     2
     4      4     68      8    -1

```

Aby zapisać wektor lub macierz w pliku korzystamy z komendy `save` podając najpierw nazwę pliku, potem nazwę zmiennej. Oto przykład prostego zastosowania tej funkcji.

```

octave:3> x=[1 943 2 13];
octave:4> save data1 x

```

MATLAB i OCTAVE mają różne domyślne typy plików, w których zapisują zmienne. W przypadku MATLABa jest to plik binarny z rozszerzeniem `.mat`, w przypadku OCTAVE – plik tekstowy ze specjalnym nagłówkiem. Szczegółowe informacje na ten temat (w tym listę dodatkowych opcji `save` pozwalających zapisać zmienne w formacie innym niż domyślny) można uzyskać pisząc `help save`. Niezależnie jednak od programu (i jego wersji), który używamy, wywołanie `load data1` w nowej sesji spowoduje, że zostanie zadeklarowana zmienna `x` a nie `data1`.

Ćwiczenie A.9 Utwórz dwie macierze `a` i `b`, np. wykorzystując funkcję `randn` i wykonaj polecenie `save data2 a b`. Co się stanie, gdy w nowej sesji MATLABa lub OCTAVE napiszesz `load data2`?

⁵ Spróbuj napisać „`help load`”.

A.4.2 Łączenie wektorów

Często zdarza się, że na bazie dwóch lub więcej wektorów chcemy utworzyć jeden duży wektor zawierający połączone dane z każdego z nich. Poniżej podajemy dwa przykłady takiej operacji.

```
octave:1> x=[1, 2, 3]; y=[4, 5, 6, 7]; [x, y]
ans =

    1    2    3    4    5    6    7
```

```
octave:2> x=[1; 2]; y=[3; 4]; [x; y]
ans =

    1
    2
    3
    4
```

Przy rozszerzaniu wektorów np. w pętli, jako początkowy wektor można przyjąć wektor pusty określony przez dwa nawiasy kwadratowe []. Połączenie wektora z wektorem pustym zwraca wektor wyjściowy.

```
octave:3> z=[]
z = [] (0x0)
octave:4> [z, x]
ans =

    1    2    3
```

Łączenie macierzy odbywa się na takich samych zasadach jak łączenie wektorów.

```
octave:5> a=ones(2); b=zeros(2,3);
octave:6> [a, b]
ans =

    1    1    0    0    0
    1    1    0    0    0
```

A.4.3 Operacje na zmiennych tekstowych

Tekst wprowadzamy zawierając go w pojedynczych cudzysłowach. Poniżej pokazujemy przykładową deklarację zmiennej tekstowej.

```
octave:1> a='funkcja exp(x) jest '
a = funkcja exp(x) jest
```

Teksty można, podobnie jak wektory, łączyć.

```
octave:2> [a, 'wypukła']
ans = funkcja exp(x) jest wypukła
```

A.4 Kilka dodatkowych możliwości programów MATLAB i OCTAVE

Do wyświetlania tekstu (także macierzy i wektorów) służy funkcja `disp`.

```
octave:3> disp(a)
funkcja exp(x) jest
```

Do wyświetlenia w trakcie wykonywania funkcji komunikatu o błędzie lub ostrzeżenia przydają się funkcje `error` i `warning`.

```
octave:4> error('To jest komunikat o bledzie')
error: To jest komunikat o bledzie
octave:4> warning('To jest ostrzezenie')
warning: To jest ostrzezenie
```

Ćwiczenie A.10 Zmień funkcję korelacja z ćwiczenia A.7, tak by wyświetlała ostrzeżenie w przypadku, gdy długości wektorów danych są różne.

A.4.4 Tworzenie wykresów

Wykresy dwuwymiarowe tworzymy przy użyciu komendy `plot`, której najczęściej podaje się dwa argumenty: wektor odciętych i wektor rzędnych punktów do umieszczenia na wykresie. Domyślnie punkty te są łączone linią ciągłą.

Oto prosty przykład wykorzystania komendy `plot`. Jako wynik uzyskujemy wykres funkcji $\cos x$ na przedziale $(-2\pi, 2\pi)$.

```
octave:1> x=(-2*pi:0.1:2*pi); y=cos(x); plot(x,y)
```

MATLAB i OCTAVE dysponują szerokim zestawem funkcji i opcji umożliwiającymi tworzenie wykresów (w tym trójwymiarowych) i ich zapis w różnego rodzaju formatach graficznych, jednak kompatybilność tych programów jest ograniczona. Za-
 interesowanego Czytelnika odsyłamy zatem do dokumentacji dostarczonej z oprogramowaniem, którego używa.

A.4.5 Sprawdzanie czasu wykonania operacji

MATLAB i OCTAVE pozwalają bardzo prosto sprawdzać i porównywać czasy (w sekundach) wykonywania różnych operacji używając słów kluczowych `tic` i `toc` według poniższego schematu.

```
tic; komendy; toc
```

Oto kilka przykładów pozwalających zobaczyć znaczenie wyboru metody dla czasu obliczeń.

Jak już wcześniej pisaliśmy pętlę są bardzo wolne w porównaniu z wbudowanymi funkcjami, takimi jak `sum`.

```
octave:1> x=rand(1,100000);
octave:2> tic; s1=sum(x); toc
ans = 0.0082500
octave:3> tic; s2=0; for i=(1:100000)
> s2=s2+x(i);
> end; toc
ans = 1.6771
```

Założmy, że chcemy policzyć sumę kwadratów liczb w wektorze wierszowym. Oto porównanie trzech podejść do tego problemu.

```
octave:4> tic; sum(x.^2); toc
ans = 0.018674
octave:5> tic; sum(x.*x); toc
ans = 0.0044490
octave:6> tic; x*x'; toc
ans = 0.0021030
```

Ćwiczenie A.11 Sprawdź ile na Twoim komputerze trwa operacja `sum(x.*x)`. Porównaj to czasem obliczeń przy wykorzystaniu pętli `for`. Jeśli masz mało pamięci operacyjnej przyjmij `x=rand(1,10000)`; zamiast `x=rand(1,100000)`;

Należy pamiętać, że czas wykonania konkretnej operacji zależy od systemu, którego używamy i obciążenia procesora innymi zadaniami.

A.4.6 Funkcja `feval`

W wielu zastosowaniach natrafiamy na problem napisania funkcji, która jako swój argument ma brać jakąś inną funkcję. Wyobraźmy sobie, że chcemy napisać funkcję, która ma rysować wykres podanej funkcji na podanym przedziale. Języki programowania takie, jak Pascal, czy C/C++ pozwalają na korzystanie z tzw. wskaźników, używając MATLAB/OCTAVE musimy korzystać z funkcji `feval`. Funkcja `feval` jako pierwszy argument bierze nazwę funkcji podaną jako tekst zaś kolejne argumenty `feval` stają się argumentami funkcji, której wartość chcemy obliczyć.

Oto kilka przykładów.

```
octave:1> feval('sin', pi/6)
ans = 0.50000
octave:2> feval('cos', pi/6)
ans = 0.86603
octave:3> feval('sum', (1:10))
ans = 55
octave:4> feval('max', [1 4 -5 7 3])
ans = 7
octave:5> feval('min', [1 4 -5 7 3])
ans = -5
```

Poniżej prezentujemy kod funkcji rysujący wykres zadanej funkcji na określonym przez użytkownika przedziale, z zadaniem odstępem między kolejnymi punktami, w których wartość funkcji jest obliczana.

Plik *example6.m*

```
function example6(funkcja, x0, x1, dx)

x=(x0:dx:x1);
y=feval(funkcja, x);
plot(x,y);
```

Koniec pliku *example6.m*

A.4 Kilka dodatkowych możliwości programów MATLAB i OCTAVE

Oto dwa przykładowe wywołania naszej funkcji.

```
octave:1> example6('sqrt', 0, 4, 0.01)
octave:2> example6('sin', -pi, pi, 0.1)
```

Ćwiczenie A.12 Przepisz funkcję `example4` (obliczającą zero funkcji $f(x) = \cos x - x$ metodą bisekcji) tak by przyjmowała nazwę funkcji i początkowy przedział ograniczający pierwiastek. Nagłówek tej funkcji powinien mieć postać:

```
function ret=bisekcja(funkcja, a, b)
```

Zapewnij kontrolę błędów, a więc sprawdź czy znaki $f(a)$ i $f(b)$ się różnią. **Wskazówka:** aby funkcja działała poprawnie w przypadkach $f(a) > 0, f(b) < 0$ i $f(a) < 0, f(b) > 0$ jako testu używaj znaku wyrażenia $f(a)f(\frac{a+b}{2})$ zamiast znaku $f(\frac{a+b}{2})$.

A.4.7 Opcjonalne argumenty funkcji. Zmienna nargin

Czasami używanie funkcji mających kilka argumentów jest uciążliwe i chcielibyśmy, aby część argumentów była opcjonalna, tzn. by w przypadku ich niepodania, program podstawił pewne domyślne wartości. Taką możliwość daje użycie zmiennej `nargin` (ang. *number of input arguments*).

Poniżej prezentujemy kod poprawionej przy użyciu `nargin` funkcji `example6` (rysującej wykresy zadanych funkcji). Użytkownik może podać albo samą nazwę funkcji (wtedy narysowany zostanie wykres na przedziale $[-1, 1]$), albo podać także przedział, albo podać także odstęp między kolejnymi punktami, w których obliczana jest wartość funkcji. Domyślna wielkość odstepu to szerokość przedziału podzielona przez 100.

Warto się przyjrzeć wykorzystaniu w tej funkcji komendy `error`. W kodzie funkcji zamieszczono dodatkowo komentarz, który można wyświetlić pisząc `help example7`

----- Plik `example7.m` -----

```
function example7(funkcja, x0, x1, dx)
%function example7(funkcja, x0, x1, dx)
%
%example7 sluzyc do rysowania wykresow zadanej funkcji
%x0, x1 - poczatek i koniec przedzialu (opcjonalne)
%dx - gestosc podzialu odcinka (x0, x1) (opcjonalne)
%
%(c) Grzegorz Klima 2004

if nargin==0
    error('Musisz podac nazwe funkcji');
end

if nargin==1
    x0=-1;
    x1=1;
end
```

```

if nargin==2
    error('Jesli podajesz poczatek przedzialu, podaj jego koniec');
end

if nargin<4
    dx=(x1-x0)/100;
end

x=(x0:dx:x1);
y=feval(funkcja, x);
plot(x,y);

```

Koniec pliku *example7.m*

Oto kilka przykładów użycia funkcji *example7*.

```

octave:1> example7()
error: Musisz podac nazwe funkcji
error: evaluating if command near line 3, column 1
error: called from 'example7' in file '/home/gklima/mfiles/example7.m'
octave:1> example7('sin')
octave:2> example7('sin',-pi)
error: Jesli podajesz poczatek przedzialu, podaj jego koniec
error: evaluating if command near line 12, column 1
error: called from 'example7' in file '/home/gklima/mfiles/example7.m'
octave:2> example7('sin',-pi,pi)
octave:3> example7('sin',-pi,pi,0.5)
octave:3> help example7
example7 is the user-defined function from the file
/home/gklima/mfiles/example7.m

```

```
function example7(funkcja, x0, x1, dx)
```

example7 służy do rysowania wykresów zadanej funkcji
x0, *x1* - początek i koniec przedziału (opcjonalne)
dx - gęstość podziału odcinka (*x0*, *x1*) (opcjonalne)

(c) Grzegorz Klima 2004

Dodatek B

Elementy metod numerycznych

Metody numeryczne są bardzo rozległą i dynamicznie rozwijającą się dziedziną matematyki stosowanej. Celem niniejszego dodatku nie jest zastąpienie lektury jakiegokolwiek podręcznika traktującego o analizie numerycznej. Zamieszczono tutaj jedynie kilka klasycznych algorytmów numerycznych i ich implementacje w języku MATLAB/OCTAVE. Wszystkie prezentowane programy są bezpośrednio wykorzystywane w książce do numerycznego rozwiązywania problemów programowania dynamicznego pojawiających się w ekonomii.

Czytelnik pragnący szerzej korzystać z metod numerycznych i głębiej je poznać powinien sięgnąć do bogatej literatury tematu. Świetnym, choć trochę leciwym podręcznikiem podstawowych metod numerycznych jest książka Ralstona [17]. Obszerną i nowoczesną prezentację metod numerycznych w ekonomii zawiera monografia Judda [11]. Miranda i Fackler [16] przedstawiają implementacje różnych algorytmów w języku MATLAB i pokazują na wielu przykładach ich zastosowanie w naukach ekonomicznych.

B.1 Podstawy. Układy równań liniowych

B.1.1 Reprezentacja zmiennoprzecinkowa liczb w komputerze

Pamięć komputerów (podobnie jak miejsce na kartce, na której zapisujemy jakąś liczbę) jest skończona. W związku z tym prawie każda liczba zapisywana jest w sposób przybliżony. Jest kilka różnych sposobów zapisywania liczb w komputerze, najpopularniejszym jest (stosowany między innymi przez programy MATLAB i OCTAVE) format zmiennoprzecinkowy. Ten sposób zapisu można w przybliżeniu przedstawić następująco (w rzeczywistości komputery używają binarnego a nie dziesiętnego formatu zapisu liczb):

$$x, \underbrace{xxx \dots x}_{n \text{ razy}} \times 10^y,$$

Dodatek B. Elementy metod numerycznych

gdzie x -y to cyfry zaś y pewien wykładnik. Jeśli np. $n = 10$, to liczba 1,0000000001 nie może zostać dokładnie zapisana i będzie reprezentowana jako 1.

W programach MATLAB i OCTAVE najmniejsza liczba ε , taka, dla której $1 + \varepsilon$ nie zostanie (błędnie) zapisane jako 1 nosi nazwę **eps**. Popatrzmy:

```
octave:1> eps
eps = 2.2204e-16
octave:2> 1+eps-1
ans = 2.2204e-16
octave:3> 1+eps/2-1
ans = 0
```

W ostatnim przypadku wynik działania $1+\text{eps}/2$ został błędnie zapisany jako 1 i zamiast uzyskać jako wynik działania $\text{eps}/2=1.1102\text{e-}16$ otrzymaliśmy 0.

Ograniczona dokładność reprezentacji liczb i potencjalne błędy, które może to powodować, sprawiają, że należy uważnie konstruować algorytmy. Wyobraźmy sobie np. zagadnienie dodania bardzo wielu liczb, wahających się od 10^{-10} do 10^{10} . Kolejność sumowania ma w takim przypadku duże znaczenie i należałoby zacząć dodawanie od liczb najmniejszych.

B.1.2 Liczba operacji

Oprócz ograniczonej dokładności drugim ważnym zagadnieniem przy korzystaniu z metod numerycznych jest liczba operacji prowadzących do wyniku. Wyobraźmy sobie problem obliczenia wartości wielomianu o współczynnikach $(a_n)_{n=0}^{N-1}$ w punkcie x :¹

$$w(x) = a_0 + a_1x + a_2x^2 + \dots + a_{N-1}x^{N-1}. \quad (\text{B-1})$$

Naturalna metoda obliczenia $w(x)$ polega na obliczeniu a_1x i dodaniu wyniku do a_0 , potem obliczeniu x^2 przemnożeniu przez a_2 i dodaniu do $a_0 + a_1x$ itd. Na obliczenie każdego składnika sumy $a_i x^i$ musimy wykonać i mnożeń, czyli łącznie potrzebujemy $N(N-1)/2$ mnożeń. Jeśli pamiętalibyśmy wynik działania x^{i-1} ($i \geq 2$) to do obliczenia $a_i x^i$ wystarczyłyby tylko 2 mnożenia, a więc minimalna zmiana algorytmu wymaga łącznie $2N - 3$ mnożeń. Okazuje się, że można tę liczbę jeszcze zmniejszyć (do $N - 1$). Wystarczy zauważyć, że:

$$w(x) = a_0 + (a_1 + (a_2x + (\dots + (a_{N-2} + a_{N-1}x)x \dots)x)x)x.$$

Jak widać, z powyższego przykładu dwa matematycznie równoważne zapisy w implementacji wiążą się z bardzo różnymi liczbami działań, a więc i łącznym czasem obliczeń.

B.1.3 Układy równań liniowych

Rozwiązywanie układów równań liniowych jest podstawowym problemem numerycznym pojawiającym się jako jeden z elementów bardzo wielu innych zagadnień

¹ Przykład ten podajemy za Juddem [11].

numerycznych. Algorytmy rozwiązywania układów równań liniowych są wkompiłowane w programy MATLAB i OCTAVE, zaś w przypadku korzystania z innych języków programowania zawsze zaleca się korzystanie z biblioteki LAPACK.² W związku z powyższym prezentacja metod rozwiązywania układów równań liniowych będzie ograniczona do podstawowych zagadnień.

Kolejne podstawienia i rozkład LU

Rozpatrzmy problem rozwiązania równania:

$$Ax = b, \quad (\text{B-2})$$

gdzie A jest macierzą $n \times n$, b danym a x szukanym wektorem kolumnowym. W przypadku, gdy A jest macierzą górno- lub dolnotrójkątną, rozwiązanie równania (B-2) jest bardzo proste. Popatrzmy na przykład, w którym A jest górnortrójkątna:

$$\begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ 0 & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_{n,n} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix} \quad (\text{B-3})$$

Jeśli (B-3) nie jest układem sprzecznym, wtedy (z ostatniego równania) mamy:

$$x_n = \frac{b_n}{a_{n,n}},$$

skąd po podstawieniu uzyskujemy:

$$x_{n-1} = \frac{b_{n-1} - a_{n-1,n}x_n}{a_{n-1,n-1}}.$$

Postępując dalej możemy wyliczyć kolejno x_{n-2} , x_{n-3} , ..., x_1 . W przypadku, gdy A jest macierzą dolnotrójkątną algorytm jest podobny. Obliczamy x_1 , potem x_2 itd.

Oczywiście najczęściej A nie jest macierzą trójkątną. Metoda eliminacji (operacji elementarnych) Gaussa pozwala sprowadzić A to takiej postaci. Niech A będzie pewną dowolną macierzą:

$$A = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1} & a_{n,2} & \cdots & a_{n,n} \end{pmatrix}.$$

Odejmując pierwszy wiersz przemnożony przez $a_{i,1}/a_{1,1}$ od i -tego ($i = 2, 3, \dots, n$) wiersza wyzerujemy wszystkie wyrazy w pierwszej kolumnie pod diagonalą. Następnie należy dla $i = 3, 4, \dots, n$ należy od i -tego wiersza odjąć wiersz drugi przemnożony

² Biblioteka LAPACK jest napisana w języku FORTRAN, jest też jej wersja w języku C. Patrz <http://www.netlib.org>.

Dodatek B. Elementy metod numerycznych

przez $a_{i,2}/a_{2,2}$. Tą metodą można wyzerować (jeśli macierz nie jest osobliwa) wszystkie elementy pod diagonalą. Jeśli nasze operacje przeprowadzalibyśmy na macierzy rozszerzonej:

$$\begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} & b_1 \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n,1} & a_{n,2} & \cdots & a_{n,n} & b_n \end{pmatrix},$$

to sprowadzilibyśmy nasz układ równań do postaci (B-3).

Operację elementarną na macierzy A polegającą na odjęciu od j -tego wiersza i -tego wiersza przemnożonego przez α można zapisać następująco:

$$BA,$$

gdzie B jest macierzą z jedynkami na diagonalu i zerami wszędzie indziej poza i -tym wierszem i j -tą kolumną, gdzie stoi $-\alpha$. By to zauważyć wystarczy zapisać B jako sumę macierzy jednostkowej i macierzy samych zer poza pozycją i, j gdzie stoi $-\alpha$ i pamiętać o rozdzielności mnożenia macierzy względem dodawania. Dalej, operacją odwrotną do odjęcia od j -tego wiersza i -tego wiersza przemnożonego przez α jest dodanie do j -tego wiersza i -tego wiersza przemnożonego przez α , któremu odpowiada macierz różniąca się od B tylko znakiem na pozycji i, j .

Niech $(B_i)_{i=1}^m$ będzie ciągiem macierzy odpowiadających wierszowym operacjom elementarnym prowadzącym do zredukowania A do postaci górnotrójkątnej. Mamy oczywistą równość:

$$B_1^{-1}B_2^{-1}\cdots B_{m-1}^{-1}B_m^{-1}B_mB_{m-1}\cdots B_2B_1 = I,$$

z której wynika:

$$B_1^{-1}B_2^{-1}\cdots B_{m-1}^{-1}B_m^{-1}B_mB_{m-1}\cdots B_2B_1A = A. \quad (\text{B-4})$$

Oznaczmy

$$L = B_1^{-1}B_2^{-1}\cdots B_{m-1}^{-1}B_m^{-1}$$

i

$$U = B_mB_{m-1}\cdots B_2B_1A.$$

Macierz U jest macierzą górnotrójkątną. Twierdzimy, że macierz L jest macierzą dolnotrójkątną.³ Mnożenie przez macierze operacji elementarnych B_i^{-1} z prawej strony sprowadza się do odpowiednich operacji nie na wierszach a na kolumnach, z tą różnicą, że jeśli jakaś macierz działa na wierszach dodając j -ty wiersz do i -tego, to na kolumnach działa dodając i -tą kolumnę do j -tej.⁴ Ponieważ macierze B_i sprowadzające A do macierzy górnotrójkątnej odpowiadają operacjom odejmowania niższego (numerem) wiersza od wyższego, operacje B_i^{-1} będą odpowiadały dodawaniu wyższej kolumny do niższej, a ponieważ L jest wynikiem działania takich operacji

³ Oznaczenia L i U pochodzą od angielskich *lower-* i *uppertriangular*.

⁴ By to dostrzec wystarczy znów przedstawić macierz B_i (czy też B_i^{-1}) jako sumę macierzy jednostkowej i macierzy z jednym wyrazem różnym od zera i skorzystać z rozdzielności mnożenia względem dodawania.

B.1 Podstawy. Układy równań liniowych

na macierzy jednostkowej, L będzie dolnotrójkątna. Inny argument sprowadza się do zauważenia, że wszystkie B_i^{-1} są dolnotrójkątne, a iloczyn macierzy dolnotrójkątnych jest macierzą dolnotrójkątną.

Mając powyższy przepis na rozkład LU zauważamy, że skoro $Ax = b$ i $A = LU$, to x jest rozwiązaniem równania $Ux = z$, gdzie z spełnia warunek $Lz = b$. Oba te równania rozwiązuje się przez kolejne podstawienia.

Należy jeszcze zwrócić uwagę na dwie potencjalne sytuacje. Po pierwsze, może się zdarzyć, że podczas redukowania A na diagonalu pojawi się 0. Wtedy, albo pod diagonalą w tej kolumnie też będą same zera, co znaczy, że macierz A jest osobliwa, albo będziemy zmuszeni do zamiany wiersza, w którym natrafiliśmy na zero na diagonalu, z innym wierszem poniżej. Tej operacji musi odpowiadać zamiana kolumn macierzy L , która będzie dalej trójkątna w tym sensie, że równanie $Lz = b$ dalej będzie można rozwiązywać podstawieniami. Po drugie, w pewnym momencie na diagonalu może pojawić się liczba mała w porównaniu z elementami poniżej i dzielenie a_{ij}/a_{jj} da w wyniku dużą liczbę, przez którą mnożymy wiersze co może prowadzić do propagacji błędu numerycznego. Wtedy zamiana wierszy też jest wskazana. Operacja ta nosi nazwę wyboru elementów podstawowych (ang. *pivoting*).

Metoda LU jest podstawową metodą numeryczną rozwiązywania równań liniowych, choć w specjalnych sytuacjach lepsze są inne rozkłady macierzy A . Poniżej pokażemy jak zaimplementowana jest metoda rozwiązywania równań liniowych w programach MATLAB i OCTAVE.

Rozwiązanie równania $Ax = b$ zapisujemy w MATLABie i OCTAVE używając znaku `\`, jak w poniższym przykładzie. Przed próbą rozwiązania równania oba programy sprawdzają najpierw, czy macierz A nie ma specjalnych właściwości ułatwiających jego rozwiązanie (np. niepotrzebny jest rozkład LU, gdy A jest trójkątna).

```
octave:1> A=randn(3); b=randn(3,1);
octave:2> x=A\b
x =
```

```
-12.2162
  4.7060
 20.7053
```

Mozemy sprawdzić jaki jest błąd naszego wyniku.

```
octave:3> A*x-b
ans =
```

```
 2.3315e-15
-1.9984e-15
 2.2204e-16
```

Należy pamiętać, aby nie zapisywać rozwiązania równania w postaci $x = \text{inv}(A) * b$ ($x = A^{-1}b$), gdyż wyznaczenie odwrotności i wykonanie mnożenia zabiera więcej obliczeń (i czasu) niż rozwiązywanie równania wprost poprzez podstawianie.⁵ Popatrzmy.

⁵ Obliczanie odwrotności A ma sens tylko wtedy, kiedy potrzebujemy rozwiązać kilka układów równań $Ax_i = b_i$.

Dodatek B. Elementy metod numerycznych

```
octave:4> A=randn(100); b=randn(100,1);
octave:5> tic; x=inv(A)*b; toc
ans = 0.0042890
octave:6> tic; x=A\b; toc
ans = 0.0022530
```

Ćwiczenie B.1 MATLAB i OCTAVE dysponują funkcją `lu` służącą do bezpośredniego wyznaczania rozkładu LU. Oblicz rozkład LU dla kilku macierzy używając tej funkcji.

Uwarunkowanie macierzy

Rozpatrzmy przykład następującego układu równań:

$$\begin{pmatrix} 1 & 1,00001 \\ 1 & 0,99999 \end{pmatrix} x = b. \quad (\text{B-5})$$

Bezpośrednim rachunkiem można sprawdzić, że gdy $b_1 = (4,00002, 3,99998)'$ rozwiązaniem jest $x_1 = (2, 2)'$, zaś gdy $b_2 = (3,9999, 4,0001)'$ rozwiązaniem jest $x_2 = (14, -10)'$. Jak widać rozwiązanie naszego układu równań jest bardzo czułe na zmianę prawej strony. Przed wprowadzeniem miary tej wrażliwości zwróćmy uwagę, że macierz naszego układu równań jest „bliska” macierzy osobliwej samych jedynek i gdyby dokładność naszego systemu zapisu liczb zmiennoprzecinkowych była poniżej 10^{-5} nasz układ równań nie byłby rozwiązywalny numerycznie.

Niech $\|\cdot\|$ będzie normą (np. euklidesową) w \mathbb{R}^n . Sensowną miarą czułości wyniku jest liczba mówiąca ile zmieni się norma wyniku jeśli norma prawej strony zmieni się o jeden procent, a więc iloraz:

$$\frac{\|x_2 - x_1\|}{\|x_1\|} / \frac{\|b_2 - b_1\|}{\|b_1\|}. \quad (\text{B-6})$$

Niech $\|A\| = \sup_{\|x\|=1} \|Ax\|$ oznacza normę macierzy (patrz dodatek C.2). Wtedy dobrym oszacowaniem ilorazu (B-6) jest uwarunkowanie macierzy określone wzorem $\text{cond } A = \|A\| \|A^{-1}\|$.

Ćwiczenie B.2 Jakie jest uwarunkowanie macierzy jednostkowej?

Sprawdźmy w MATLAB lub OCTAVE uwarunkowanie macierzy w naszym wyjściowym równaniu.

```
octave:1> a=[1 1.00001; 1 0.99999];
octave:2> cond(a)
ans = 200000.000006024
```

Dlaczego uwarunkowanie jest takie złe? Wystarczy sprawdzić jaka jest odwrotność naszej macierzy.

```
octave:3> inv(a)
ans =

-49999.4999999500    50000.4999999500
 49999.9999999500   -49999.9999999500
```

Jakie będą numeryczne rozwiązania naszego układu?⁶

```
octave:4> a\[4.00002 3.99998]’
ans =
```

```
1.99999999998890
2.00000000001110
```

```
octave:5> a\[3.9999 4.0001]’
ans =
```

```
13.99999999998890
-9.99999999998890
```

Widać, że w przypadku, w którym dokładność reprezentacji jest dużo lepsza niż uwarunkowanie problemu, nawet gdy to ostatnie jest duże, błąd otrzymanego wyniku jest relatywnie mały.

Ćwiczenie B.3 Przykładem zawsze nieosobliwej, lecz bardzo źle uwarunkowanej macierzy jest tzw. macierz Hilberta $H_{i,j} = 1/(i+j-1)$. Używając funkcji `hilb` sprawdź uwarunkowanie macierzy Hilberta dla rzędów 2, 3, 5, 10, 20, 30. Spróbuj obliczyć H^{-1} korzystając z polecenia `inv` a następnie sprawdź jak `hilb(n)*inv(hilb(n))` różni się od macierzy jednostkowej dla kilku n . Dla jakiego n `norm(hilb(n)*inv(hilb(n))-eye(n))` jest większe od 0.001?

B.2 Aproksymacja funkcji. Wielomiany Czebyszewa

7

B.2.1 Podstawowy problem aproksymacji wielomianami

Rozpatrzmy problem aproksymacji wielomianami pewnej ciągłej funkcji $f(x)$ na przedziale $[a, b]$. Załóżmy, że znamy wartość funkcji f w pewnych punktach $a \leq x_1 < x_2 < \dots < x_n \leq b$. Jeśli przyjmiemy, że stopień wielomianu aproksymującego ma być nie większy niż $n - 1$, wtedy można tak dobrać współczynniki α_i , aby:

$$\alpha_0 + \alpha_1 x_i + \dots + \alpha_{n-1} x_i^{n-1} = f(x_i).$$

Zauważmy, że problem doboru wektora współczynników $\alpha = (\alpha_i)_{i=0}^{n-1}$ sprowadza się do rozwiązania równania:

$$\begin{pmatrix} 1 & x_1 & \dots & x_1^{(n-1)} \\ 1 & x_2 & \dots & x_2^{(n-1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & \dots & x_n^{(n-1)} \end{pmatrix} \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_{n-1} \end{pmatrix} = \begin{pmatrix} f(x_1) \\ f(x_2) \\ \vdots \\ f(x_n) \end{pmatrix}. \quad (\text{B-7})$$

⁶ W dalszym ciągu, ze względu na fakt, iż interesują nas potencjalnie małe błędy numeryczne (na dalekich miejscach po przecinku) używamy formatu wydruku, który można uzyskać wydając polecenie `format long`.

Dodatek B. Elementy metod numerycznych

Macierze postaci $V_{i,j} = x_i^{j-1}$ noszą nazwę macierzy Vandermonde'a. Macierze takie są odwracalne, tj. zawsze istnieje rozwiązanie równania (B-7). Niestety, w zależności od precyzji reprezentacji liczb zmiennoprzecinkowych w naszym komputerze, od pewnego n macierze Vandermonde'a stają się nieodwracalne numerycznie, co jest konsekwencją ich złego uwarunkowania.⁷

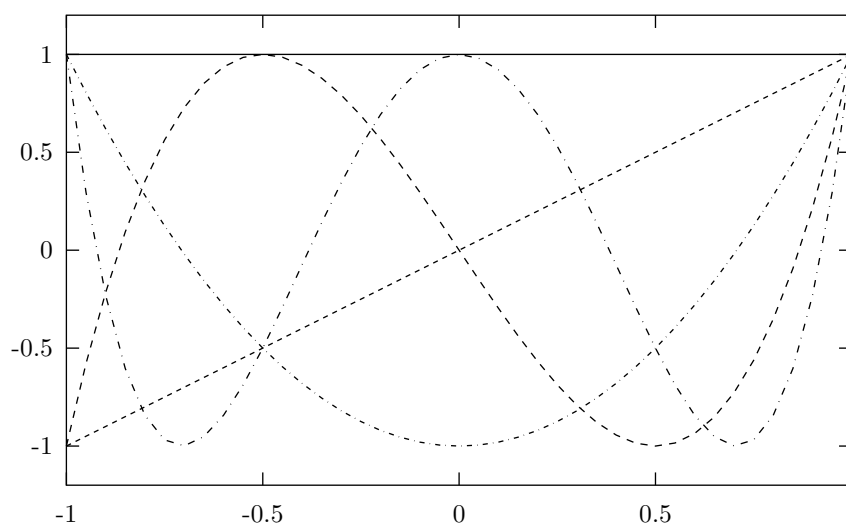
Ćwiczenie B.4 Przyjmij węzły x_i w punktach $0; 0,1; 0,2; \dots; 2$ ($n = 21$). Utwórz macierz Vandermonde'a dla tych węzłów i spróbuj ją odwrócić numerycznie w MATLABie lub OCTAVE. Czy w wyniku działania $A \cdot \text{inv}(A)$ uzyskuje się macierz jednostkową?

B.2.2 Dwie definicje wielomianów Czebyszewa

Okazuje się, że w zagadnieniach aproksymacji i pochodnych zagadnieniach numerycznych (np. kwadratury) o wiele lepiej jest zamiast z bazy $(1, x, x^2, \dots, x^n)$ korzystać z bazy wielomianów ortogonalnych, w szczególności z wielomianów Czebyszewa. Poniżej podajemy ich definicję. Rysunek B.1 przedstawia wykres pierwszych pięciu wielomianów Czebyszewa.

Definicja B.1 Wielomianem Czebyszewa n -tego rzędu na przedziale $[-1, 1]$ nazywamy funkcję:

$$T_n(x) = \cos(n \arccos(x)). \quad (\text{B-8})$$



Rysunek B.1: Pierwsze pięć wielomianów Czebyszewa

Ćwiczenie B.5 Pokaż, że wielomiany Czebyszewa tworzą rodzinę ortogonalną z funkcją wagową $\omega(x) = 1/\sqrt{1-x^2}$, tzn.

$$\int_{-1}^1 \omega(x) T_n(x) T_m(x) dx \neq 0 \iff m \neq n. \quad (\text{B-9})$$

Wskazówka: dokonaj w całce podstawienia $t = \arccos(x)$.

⁷ Wektor α można wyznaczyć inaczej, korzystając ze wzoru interpolacyjnego Lagrange'a. Nie będziemy tu rozwijać tej myśli. Zainteresowanego Czytelnika odsyłamy do Ralstona [17] i Judda [11].

W powyższej definicji wielomiany Czebyszewa są określone na przedziale $[-1, 1]$ a więc użyteczne tylko w zagadnieniach aproksymacji na tym samym przedziale. Aby nasze nowe narzędzie było użyteczne w ogólnym przypadku wystarczy dokonać liniowej transformacji z przedziału $[a, b]$ na przedział $[-1, 1]$ zgodnie ze wzorem:

$$z = 2\frac{x-a}{b-a} - 1. \quad (\text{B-10})$$

Ćwiczenie B.6 Jak wygląda transformacja odwrotna?

W obliczeniach numerycznych korzysta się z innej (rekurencyjnej) definicji wielomianów Czebyszewa. Jest ona zawarta w poniższym fakcie (przy okazji pokazujemy, że $T_n(x)$ jest rzeczywiście wielomianem).

Fakt B.1 *Spełnione są następujące związki:*

$$T_0(x) = 1, \quad (\text{B-11})$$

$$T_1(x) = x, \quad (\text{B-12})$$

$$T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x). \quad (\text{B-13})$$

Dowód. Pierwsze dwie równości są trywialne. Trzecia wynika z własności funkcji trygonometrycznych. Oznaczmy $y = \arccos(x)$. Mamy:

$$\begin{aligned} T_n(x) &= \cos(ny) = \cos((n-1)y + y) = \\ &= \cos((n-1)y)\cos(y) - \sin((n-1)y)\sin(y) = \\ &= xT_{n-1}(x) - \sin((n-2)y + y)\sin(y) = \\ &= xT_{n-1}(x) - (\sin((n-2)y)\cos(y) + \cos((n-2)y)\sin(y))\sin(y) = \\ &= xT_{n-1}(x) - \sin((n-2)y)\cos(y)\sin(y) - \cos((n-2)y)\sin^2(y) = \\ &= xT_{n-1}(x) - \sin((n-2)y)\cos(y)\sin(y) - \cos((n-2)y) + \cos((n-2)y)\cos^2(y) = \\ &= xT_{n-1}(x) - T_{n-2}(x) + (\cos((n-2)y)\cos(y) - \sin((n-2)y)\sin(y))\cos(y) = \\ &= xT_{n-1}(x) - T_{n-2}(x) + x\cos((n-1)y) = 2xT_{n-1}(x) - T_{n-2}(x). \quad \blacksquare \end{aligned}$$

Aproksymując funkcję przy wykorzystaniu wielomianów Czebyszewa będziemy spotykali się z zagadnieniem obliczenia poniższego szeregu Czebyszewa:

$$\sum_{i=0}^{n-1} \alpha_i T_i(x). \quad (\text{B-14})$$

Poniższa funkcja MATLAB/OCTAVE oblicza wartość (B-14) dla podanych α i x korzystając z faktu B.1. Przyjmujemy konwencję, że wektor współczynników jest wektorem kolumnowym i (inaczej niż w konwencji MATLAB/OCTAVE) współczynniki są uporządkowane $\alpha_0, \dots, \alpha_{n-1}$. Funkcja jest tak napisana, że wylicza wartość wielomianu Czebyszewa od razu dla całego wektora (kolumnowego) \mathbf{x} wartości x . Opcjonalnymi argumentami są początek i koniec przedziału, na którym szereg (B-14) jest określony.

Dodatek B. Elementy metod numerycznych

```

Plik chebval.m
function ret=chebval(alpha, x, a, b)
%function ret=chebval(alpha, x, a, b)
%
%chebval oblicza wartosc szeregu Czebyszewa
%alpha - wspolczynniki przy Tn(x) n=0,1,2,.. (wektor kolumnowy)
%x - punkty, w ktorych ma byc obliczony szereg (wektor kolumnowy)
%a, b (opcjonalne) - poczatek i koniec przedzialu, ktorym
%szereg jest okreslony, domyslnie a=-1, b=1
%
%(c) Grzegorz Klima 2004

if nargin<4
    a=-1;
    b=1;
end

x=2*(x-a)/(b-a)-1; %normalizacja

n=size(alpha, 1);
m=size(x, 1);

t=ones(m, n);

t(:,2)=x;

for i=(3:n)
    t(:,i)=2*x.*t(:,i-1)-t(:,i-2);
end

ret=t*alpha;
Koniec pliku chebval.m

```

Ćwiczenie B.7 Korzystając z funkcji `chebval` zrób wykresy takie, jak na rysunku B.1.

B.2.3 Węzły Czebyszewa

Do tej pory przemilczeliśmy kwestię doboru węzłów, w których obliczamy wartość aproksymowanej funkcji f . Jak dowodzi teoria i pokazuje doświadczenie numeryczne, „naturalne” podejście polegające na podzieleniu przedziału na $n+1$ równych części nie jest najlepszą strategią. Korzystając z wielomianów Czebyszewa najlepiej wybierać węzły w punktach określonych jak niżej.

Definicja B.2 Węzłami Czebyszewa na przedziale $[-1, 1]$ nazywamy punkty ($i = 1, \dots, n$):

$$x_i = \cos\left(\frac{n-i+\frac{1}{2}\pi}{n}\right). \quad (\text{B-15})$$

Ćwiczenie B.8 Pokaż, że węzły Czebyszewa są miejscami zerowymi $T_n(x)$.

Ćwiczenie B.9 Jak zdefiniować węzły Czebyszewa na arbitralnym przedziale $[a, b]$?

Zauważmy, że węzły Czebyszewa są bardziej skoncentrowane przy granicach przedziału, co istotnie zmniejsza błąd aproksymacji, który w przypadku równomiernego rozłożenia węzłów może (nawet dla relatywnie gładkich funkcji) przy końcach przedziału wzrastać wielokrotnie.

Poniżej prezentujemy funkcję MATLAB/OCTAVE wyznaczającą węzły Czebyszewa dla zadanych n , a , i b . W sytuacji gdy podany jest tylko jeden argument (n) funkcja zwraca węzły na $[-1, 1]$.

Plik *chebnod.m*

```
function ret=chebnod(n,a,b)
%function ret=chebnod(n,a,b)
%
%chebnod zwraca wektor kolumnowy n wezlow Czebyszewa na a,b
%domyslnie a=-1, b=1;
%
%(c) Grzegorz Klima 2004

if nargin<3
    a=-1;
    b=1;
end

x=(a+b)/2+(b-a)*cos(pi*((n-1:-1:0)+.5)/n)/2;
ret=x';
```

Koniec pliku *chebnod.m*

Jeśli teraz chcielibyśmy dobrać współczynniki α tak aby $\sum_{i=0}^{n-1} \alpha_i T_i(x_j) = f(x_j)$, gdzie x_j są węzłami Czebyszewa, to równanie określające wektor α jest podobne do równania (B-7):

$$\begin{pmatrix} T_0(x_1) & T_1(x_1) & \cdots & T_{n-1}(x_1) \\ T_0(x_2) & T_1(x_2) & \cdots & T_{n-1}(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ T_0(x_n) & T_1(x_n) & \cdots & T_{n-1}(x_n) \end{pmatrix} \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_n \end{pmatrix} = \begin{pmatrix} f(x_1) \\ f(x_2) \\ \vdots \\ f(x_n) \end{pmatrix}. \quad (\text{B-16})$$

Macierz $\Phi_{i,j} = T_{j-1}(x_i)$ różni się jednak zasadniczo w swych własnościach od macierzy Vandermonde'a. Jest bardzo dobrze uwarunkowana i to niezależnie od n . Proste podstawienie pozwala stwierdzić, że:

$$\Phi_{i,j} = \cos\left(\frac{n-i+\frac{1}{2}}{n}\pi(j-1)\right). \quad (\text{B-17})$$

Zauważmy, że macierz Φ nie musi być dobierana do przedziału $[a, b]$, na którym aproksymujemy funkcję f , bo obliczając jej elementy najpierw przy wyliczaniu węzłów transformujemy z -ty z $[-1, 1]$ na x -y z przedziału $[a, b]$, a potem obliczając $T(x)$ transformujemy x -y (węzły) z powrotem na przedział $[-1, 1]$.

Poniżej podajemy funkcję MATLAB/OCTAVE zwracającą macierz Φ dla danego n . Warto zrozumieć, w jaki sposób macierz Φ jest w tej funkcji obliczona. Korzystamy tutaj z funkcji `diag`, która dla danego wektora zwraca macierz z tym wektorem na diagonalu.

Dodatek B. Elementy metod numerycznych

Plik *chebphi.m*

```
function ret=chebphi(n)
%function ret=chebphi(n)
%
%chebphi zwraca macierz wartosci wielomianow Czebyszewa
%w wezlach Czebyszewa
%
%(c) Grzegorz Klima 2004

ret=cos((n-.5:-1:0.5)')*ones(1,n)*diag((0:n-1))*pi/n;
Koniec pliku chebphi.m
```

Pokażemy teraz jak wykorzystać podane funkcje do przybliżenia funkcji $f(x) = xe^{-x}$ na przedziale $[0, 5]$. Przyjmujemy $n = 10$. Zacniemy od wyznaczenia węzłów na $[0, 5]$.

```
octave:1> w=chebnd(10, 0, 5);
```

Następnie obliczymy wartość funkcji f w tych węzłach.

```
octave:2> f=w.*exp(-w);
```

Pozostaje tylko rozwiązać układ równań (B-16).

```
octave:3> a=chebphi(10) \ f;
```

Mając wyznaczone współczynniki, możemy obliczyć wartości szeregu aproksymującego f na zadanym przedziale i np. narysować wykres naszej aproksymacji.

```
octave:4> x=(0:0.1:5)'; y=chebval(a, x, 0, 5);
```

```
octave:5> plot(x,y)
```

B.3 Rozwiązywanie równań nieliniowych

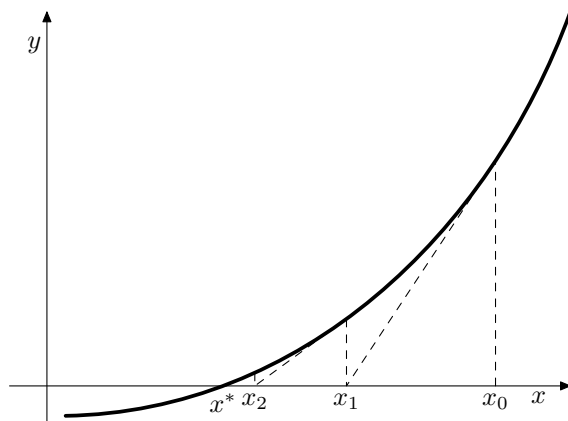
W dodatku A omówiliśmy najprostszą metodę znajdowania zer funkcji $f: \mathbb{R} \rightarrow \mathbb{R}$ zwaną metodą bisekcji. Poniżej przedstawimy dwie bardzo popularne metody rozwiązywania równań nieliniowych: metodę Newtona i metodę siecznych i ich uogólnienia na przypadek wielowymiarowy. Oczywiście, jest kilka innych metod rozwiązywania układów nieliniowych, które w pewnych sytuacjach mogą okazać się dużo lepsze (choć trudniejsze w implementacji) niż te przedstawione poniżej. Szeroki przegląd metod numerycznych dla układów równań nieliniowych można znaleźć u Judda [11].

B.3.1 Równania nieliniowe w jednym wymiarze

Metoda Newtona

Idea metody Newtona przedstawiona jest na rysunku B.2. Załóżmy, że startujemy z punktu x_0 jako pierwszego przybliżenia naszego pierwiastka. Jeśli $f'(x_0) \neq 0$, wtedy styczna do wykresu funkcji w punkcie $(x_0, f(x_0))$ przecina oś x -ów w pewnym

punkcie x_1 , który bierzemy jako nowe przybliżenie szukanego pierwiastka. W każdej iteracji mając x_n , kreślimy styczną do wykresu f w punkcie $(x_n, f(x_n))$ i ustalamy nowe przybliżenie x_{n+1} w punkcie jej przecięcia z osią x . Procedurę kończymy, gdy uzyskamy satysfakcjonujące przybliżenie, tj. gdy $|f(x_n)| < \varepsilon$, gdzie ε jest pewnym ustalonym przez nas poziomem tolerancji.



Rysunek B.2: Metoda Newtona znajdowania pierwiastka $f(x^*) = 0$

Równanie stycznej przechodzącej przez punkt $(x_n, f(x_n))$ ma postać:

$$y = f(x_n) + f'(x_n)(x - x_n)$$

Przyrównując $y = 0$ uzyskujemy regułę iteracyjną dla metody Newtona:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}. \quad (\text{B-18})$$

Poniżej prezentujemy prosty kod w języku MATLAB/OCTAVE wyznaczający $\sqrt[3]{2}$ metodą Newtona zastosowaną do funkcji $f(x) = x^3 - 2$ z punktem startowym $x_0 = 2$.

```
octave:2> x=2;
octave:3> while abs(x^3-2)>eps
> x=x-(x^3-2)/(3*x^2)
> end
x = 1.5000000000000000
x = 1.29629629629630
x = 1.26093222474175
x = 1.25992186056593
x = 1.25992104989539
x = 1.25992104989487
```

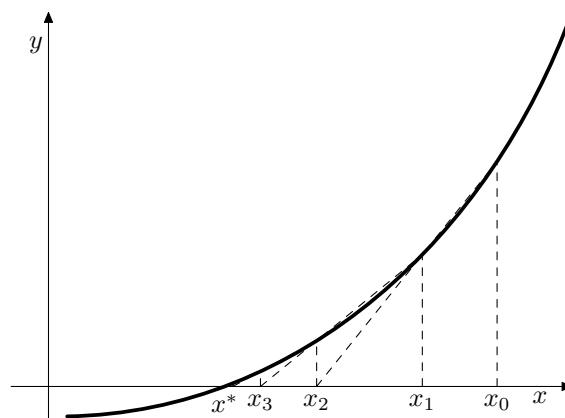
Jak widać zbieżność metody Newtona (jeśli metoda ta zbiega) jest bardzo dobra.

Zbieżność metody Newtona jest zagwarantowana tylko gdy funkcja f jest wypukła (wklęsła) lub startujemy odpowiednio blisko pierwiastka x^* i $|f''(x^*)/f'(x^*)|$ jest skończone.

Metoda Newtona wymaga obliczenia wartości pochodnej w każdym punkcie iteracji. Opisana poniżej metoda siecznych omija ten problem.

Metoda siecznych

Metoda siecznych wychodzi od dwóch punktów startowych (przybliżeń pierwiastka) x_0 i x_1 . Trzecie przybliżenie pierwiastka (x_2) uzyskuje się przeprowadzając sieczną przez punkty $(x_0, f(x_0))$ i $(x_1, f(x_1))$ i znajdując punkt jej przecięcia z osią x -ów. W kolejnych iteracjach mając x_n i x_{n-1} przybliżenie x_{n+1} wyznaczamy znajdując przecięcie siecznej przechodzącej przez $(x_{n-1}, f(x_{n-1}))$ i $(x_n, f(x_n))$ z osią x -ów. Metoda siecznych przedstawiona jest na rysunku B.3.



Rysunek B.3: Metoda siecznych znajdowania pierwiastka $f(x^*) = 0$

Równanie siecznej przechodzącej przez punkty $(x_{n-1}, f(x_{n-1}))$ i $(x_n, f(x_n))$ to:

$$y = f(x_n) + \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}(x - x_n),$$

skąd (po przyjęciu $y = 0$) wynika reguła iteracyjna dla metody siecznych:

$$x_{n+1} = x_n - \frac{f(x_n)(x_n - x_{n-1})}{f(x_n) - f(x_{n-1})}. \quad (\text{B-19})$$

Poniższy kod wyznacza $\sqrt[3]{2}$ metodą siecznych.

```
octave:1> x0=3; x1=2;
octave:2> while abs(x1^3-2)>eps
> x=x1-(x1^3-2)*(x1-x0)/(x1^3-x0^3);
> x0=x1;
> x1=x
> end
x1 = 1.68421052631579
x1 = 1.41205211726384
x1 = 1.29892714350933
x1 = 1.26419346754830
x1 = 1.26005034713575
x1 = 1.25992148732506
x1 = 1.25992104993976
x1 = 1.25992104989487
```

B.3 Rozwiązywanie równań nieliniowych

Jak widać metoda siecznych jest wolniejsza od metody Newtona. Jest to koszt związany z uniknięciem obliczania pochodnej f .

Przyjrzyjmy się na koniec związkom metody siecznych i Newtona. Podstawmy w równaniu (B-19)

$$a_n = \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}.$$

Pozwala to przepisać (B-19) następująco:

$$x_{n+1} = x_n - \frac{f(x_n)}{a_n}.$$

Dalej przyjmijmy:

$$\begin{aligned} a_{n+1} &= a_n - \frac{f(x_{n+1})}{f(x_n)} a_n = -\frac{f(x_{n+1}) - f(x_n)}{f(x_n)} a_n = \\ &= \frac{f(x_{n+1}) - f(x_n)}{f(x_n)} \frac{f(x_n)}{x_{n+1} - x_n} = \frac{f(x_{n+1}) - f(x_n)}{x_{n+1} - x_n}. \end{aligned}$$

Przy tak określonym a_{n+1} mamy:

$$x_{n+2} = x_{n+1} - \frac{f(x_{n+1})}{a_{n+1}}.$$

Powyższe przekształcenia pozwalają zinterpretować metodę siecznych jako zmodyfikowaną metodę Newtona ze schematem rekurencyjnym

$$x_{n+1} = x_n - \frac{f(x_n)}{a_n}, \quad (\text{B-20})$$

w którym pochodna $f'(x_n)$ jest zastąpiona pewnym przybliżeniem a_n , poprawianym w kolejnych iteracjach według formuły:

$$a_{n+1} = a_n - \frac{f(x_{n+1})}{f(x_n)} a_n. \quad (\text{B-21})$$

Ćwiczenie B.10 Znajdź $\sqrt[3]{2}$ korzystając z powyższego sformułowania metody siecznych. Przyjmij $x_0 = 2$. Sprawdź zachowanie metody dla kolejno: $a_0 = 19 (= (f(3) - f(2))/(3 - 2))$, $a_0 = 12 (= f'(2))$ i $a_0 = 1$.

B.3.2 Układy równań nieliniowych

Metoda Newtona

Niech $f: \mathbb{R}^k \rightarrow \mathbb{R}^k$. Metoda Newtona znalezienia pierwiastka równania $f(x) = 0$ w takim przypadku sprowadza się do następującego schematu iteracyjnego (przyjmujemy, że x i $f(x)$ są wektorami kolumnowymi):

$$x_{n+1} = x_n - J|_{x=x_n}^{-1} f(x_n), \quad (\text{B-22})$$

Dodatek B. Elementy metod numerycznych

gdzie J oznacza macierz Jacobiego pochodnych cząstkowych

$$J_{i,j}|_{x=x_n} = \frac{\partial f^i(x_1, \dots, x_k)}{\partial x_j} \Big|_{x=x_n}.$$

Uzasadnienie wzoru (B-22) jest podobne do uzasadnienia (B-18). Ze wzoru Taylora dla funkcji z \mathbb{R}^k w \mathbb{R}^k mamy:

$$f(x) \approx f(x^n) + J|_{x=x_n}(x - x^n).$$

Przyjmując w powyższym (przybliżonym) równaniu $f(x) = 0$ uzyskujemy (B-22).

Przed podaniem kodu funkcji rozwiązującej układ równań nieliniowych metodą Newtona musimy odnieść się do kwestii określenia macierzy $J_{i,j}$. Ręczne różniczkowanie f i zakodowanie funkcji zwracającej macierz Jacobiego w większości zastosowań jest procesem bardzo uciążliwym i obciążonym dużym ryzykiem błędu.⁸ W naszej implementacji metody Newtona będziemy się odwoływać do funkcji obliczającej pochodne numerycznie. Skorzystamy z następującego przybliżenia pochodnej (tzw. metody różnic dwustronnych):⁹

$$J_{i,j} = \frac{\partial f^i(x_1, \dots, x_k)}{\partial x_j} \approx \frac{f^i(x_1, \dots, x_j + h, \dots, x_k) - f^i(x_1, \dots, x_j - h, \dots, x_k)}{2h}.$$

Poniżej podajemy kod funkcji `jacob` wyznaczającej macierz Jacobiego numerycznie. W implementacji przyjmujemy h na poziomie pierwiastka trzeciego stopnia z dokładności reprezentacji liczb zmiennoprzecinkowych w naszym systemie; dodatkowo zamiast dzielić przyrost funkcji przez $2h$ dzielimy go przez $(x_j + h) - (x_j - h)$ dla minimalizacji błędu związanego z dodawaniem i odejmowaniem potencjalnie małego w stosunku do x_j h .¹⁰ Funkcja `jacob` w jednej iteracji wyznacza całą kolumnę macierzy J .

Plik `jacob.m`

```
function ret=jacob(fun, x)
%function ret=jacob(fun, x)
%
%jacob oblicza numeryczne przyblizenie macierzy Jacobiego
%funkcji fun w punkcie x metoda roznic dwustronnych
%
%(c) Grzegorz Klima 2004

m=size(x,1);

h=eps^(1/3);

jac=[];
```

⁸ Jednym z potencjalnych rozwiązań tego problemu jest wykorzystanie jednego z programów obliczeń symbolicznych i przekonwertowanie wyniku różniczkowania symbolicznego na funkcję MATLAB/OCTAVE.

⁹ Patrz np. Ralston [17].

¹⁰ Szersze uzasadnienie tych reguł Czytelnik znajdzie u Mirandy i Facklera [16].

B.3 Rozwiązywanie równań nieliniowych

```

for j=(1:m)
    x1=x;
    x2=x;
    x1(j)=x(j)+h;
    x2(j)=x(j)-h;
    f1=feval(fun, x1);
    f2=feval(fun, x2);
    jac=[ jac, (f1-f2)/(x1(j)-x2(j)) ];
end

ret=jac;

```

————— Koniec pliku *jacob.m* —————

Mając funkcję obliczającą numerycznie macierz pochodnych możemy przystąpić do zaimplementowania metody Newtona.

————— Plik *newton.m* —————

```

function sol=newton(fun, x0, maxiter, tol)
%function sol=newton(fun, x0, maxiter, tol)
%
%funkcja newton rozwiazuje ukklad n rownan nieliniowych
%z n niewiadomymi metoda Newtona
%
%fun - funkcja, ktorej zero ma byc znalezione
%x0 - punkt starowy iteracji
%maxiter (opcjonalne) - max liczba iteracji (dom. 100)
%tol - poziom tolerancji (dom. eps)
%
%(c) Grzegorz Klima 2004

if nargin<4
    tol=eps;
end;

if nargin<3
    maxiter=100;
end;

x=x0;
n=0;
f=feval(fun, x);

while (f'*f>tol)&(n<maxiter)
    j=jacob(fun, x);
    dx=-j\f;
    x=x+dx;
    f=feval(fun, x);
    n=n+1;
end

if (f'*f>tol)

```

Dodatek B. Elementy metod numerycznych

```

warning('Funkcja newton nie zbiegla do rozwiazania')
end

sol=x;

```

Koniec pliku *newton.m*

Zastosujemy funkcję `newton` do rozwiązania następującego układu równań (jest to układ równań określający stan ustalony modelu Solowa wyznaczany przez złotą regułę):

$$\begin{cases} sf(k) = \delta k \\ f'(k) = \delta \end{cases},$$

gdzie $f(k) = \sqrt[3]{k}$ i $\delta = 0,02$.

W poniższym kodzie $x(1)$ odpowiada s , $x(2)$ odpowiada k , $y(1)$ pierwszemu, zaś $y(2)$ drugiemu równaniu.

Plik *solow.m*

```

function y=solow(x)

y=zeros(2,1);
y(1)=x(1)*x(2)^(1/3)-.02*x(2);
y(2)=1/3*x(2)^(-2/3)-.02;

```

Koniec pliku *solow.m*

Możemy teraz wywołać funkcję `newton` aby uzyskać rozwiązanie. Jako punkt startowy przyjmujemy $k = 1$ i $s = 1/2$.

```

octave:2> newton ('solow', [0.5 1]')
ans =

```

```

0.333333331751857
68.041381179943443

```

Metoda Broydena

Metoda Broydena jest uogólnieniem metody siecznych w wersji opisanej przez równania (B-20)-(B-21). Uogólnienie to jest jednak mniej oczywiste niż w przypadku metody Newtona. Zauważmy, że w przypadku ogólnym ($f: \mathbb{R}^k \rightarrow \mathbb{R}^k$) równanie

$$f(y) - f(x) = A(y - x),$$

ma wiele rozwiązań A , gdyż jest to układ k równań liniowych z k^2 niewiadomymi. Stwarza to problem określenia metody poprawiania A , która ma być przybliżeniem macierzy Jacobiego. Idea metody Broydena jest taka, by dokonywać najmniejszej możliwej poprawki zgodnej z warunkiem siecznej, tj. A_{n+1} musi spełniać

$$f(x_{n+1}) - f(x_n) = A_{n+1}(x_{n+1} - x_n)$$

i jednocześnie być bliska A_n . Formułę poprawki Broydena podajemy bez formalnej definicji, co znaczy „blisko” w odniesieniu do macierzy i bez wyprowadzenia. Oto ona:

$$A_{n+1} = A_n + \frac{[f(x_{n+1}) - f(x_n) - A_n(x_{n+1} - x_n)](x_{n+1} - x_n)'}{(x_{n+1} - x_n)'(x_{n+1} - x_n)}. \quad (\text{B-23})$$

B.3 Rozwiązywanie równań nieliniowych

Kolejne przybliżenie pierwiastka jest obliczane w naturalny sposób:

$$x_{n+1} = x_n - A_n^{-1} f(x_n). \quad (\text{B-24})$$

Poniżej podajemy prostą implementację metody Broydena.¹¹ Jako pierwsze przybliżenie macierzy Jacobiego A_0 przyjmujemy numeryczną pochodną w x_0 obliczoną dzięki odwołaniu do funkcji `jacob`.

Plik *broyden.m*

```
function sol=broyden(fun, x0, maxiter, tol)
%function sol=broyden(fun, x0, maxiter, tol)
%
%funkcja broyden rozwiazuje ukklad n rownan nieliniowych
%z n niewiadomymi metoda Broydena
%
%fun - funkcja, ktorej zero ma byc znalezione
%x0 - punkt starowy iteracji
%maxiter (opcjonalne) - max liczba iteracji (dom. 100)
%tol - poziom tolerancji (dom. eps)
%
%(c) Grzegorz Klima 2004

if nargin<4
    tol=eps;
end;

if nargin<3
    maxiter=100;
end;

x=x0;
n=0;
f=feval(fun, x);
j=jacob(fun, x);

while (f'*f>tol)&(n<maxiter)
    dx=-j\f;
    x=x+dx;
    fold=f;
    f=feval(fun, x);
    df=f-fold;
    j=j+(df-j*dx)*dx'/(dx'*dx);
    n=n+1;
end

if (f'*f>tol)
    warning('Funkcja broyden nie zbiegla do rozwiazania')
end
```

¹¹ Metodę Broydena można ulepszyć dokonując przybliżenia nie samej macierzy Jacobiego tylko jej odwrotności. Takie podejście omija problem rozwiązywania układu równań liniowych.

Dodatek B. Elementy metod numerycznych

```
sol=x;
```

Koniec pliku *broyden.m*

B.3.3 Kilka uwag na temat rozwiązywania równań nieliniowych

O metodach numerycznych mówi się, iż są jednocześnie nauką i sztuką. Nauka podpowiada pewne algorytmy, pozwala określić warunki zbieżności i jej tempo. W większości przypadków albo założenia odpowiednich twierdzeń dotyczących zbieżności algorytmów nie są spełnione, albo też nie jesteśmy w stanie tego sprawdzić (skąd mamy wiedzieć, że nasz punkt startowy jest w otoczeniu pierwiastka, skoro go nie znamy?). Korzystając z metod takich jak te opisane powyżej (niegwarantujących globalnej zbieżności) należy pamiętać o tych problemach. W szczególności, jeśli nasza metoda nie zbiega trzeba spróbować znaleźć lepszy punkt startowy. Podpowiedzi może dostarczyć albo struktura matematyczna problemu, albo intuicja lub wiedza na temat własności rozwiązania. Jeśli mamy do czynienia z zagadnieniami ekonomicznymi, to trudno oczekiwać zbieżności, gdy wystartujemy ze stopą oszczędności powyżej 1 lub ujemną stopą procentową. Często sprawdza się heurystyczna metoda polegająca na uprzednim rozwiązaniu prostszego problemu i potraktowaniu wyniku jako punktu startowego w problemie trudniejszym.¹²

W problemach równań nieliniowych i optymalizacji w przypadku, gdy nie stosujemy (np. ze względu na kwestie prędkości czy też problemy z implementacją) algorytmów zapewniających globalną zbieżność, skuteczność jest mocno skorelowana z doświadczeniem i zdrowym rozsądkiem użytkownika.

Dodatek C

Podstawy analizy funkcjonalnej

Analizę funkcjonalną można najkrócej (choć z dużym uproszczeniem) określić jako dział matematyki uogólniający wyniki klasycznej (uprawianej w przestrzeniach \mathbb{R}^n) analizy, algebry liniowej i geometrii. Najważniejsze analogie budowane są przez uogólnienia standardowych pojęć wartości bezwzględnej, długości, odległości, operacji liniowej i pochodnej. Poniżej prezentujemy podstawowe wyniki analizy funkcjonalnej wykorzystywane w toku wyvodu oraz pewne jej zastosowania rozszerzające dyskusję z niektórych rozdziałów podręcznika. Prezentacja ma bardzo „geometryczny” charakter, co powinno uczynić ją przystępną dla Czytelnika bez wcześniejszego przygotowania z zakresu teorii mnogości i topologii. Rozszerzenie poniższych treści zawierają m.in. podręczniki (adresowane do fizyków i matematyków) Maurina [15] i Rudina [19], [20].

C.1 Przestrzenie Banacha

C.1.1 Przestrzenie liniowe

Zacznijmy od podstawowej definicji.

Definicja C.1 *Zbiór X elementów x, y, z, \dots zwanych wektorami nazywamy przestrzenią liniową nad ciałem skalarów \mathbb{R} (lub ciałem liczb zespolonych \mathbb{C}) jeśli określone są w niej działania dodawania wektorów i mnożenia wektorów przez skalary spełniające są poniższe warunki:*

– *warunek przemienności dodawania wektorów:*

$$x + y = y + x, \tag{C-1}$$

– *warunek łączności dodawania wektorów:*

$$(x + y) + z = y + (x + z), \tag{C-2}$$

– *istnieje wektor zerowy $0 \in X$ (element neutralny dodawania wektorów), który spełnia*

$$\forall_{x \in X} x + 0 = x, \tag{C-3}$$

Dodatek C. Podstawy analizy funkcjonalnej

– dla każdego $x \in X$ istnieje element przeciwny $(-x)$, taki, że:

$$x + (-x) = 0, \quad (\text{C-4})$$

– działania dodawania wektorów i dodawania skalarów są rozdzielne względem mnożenia przez skalar, tj.:

$$\alpha(x + y) = (\alpha x) + (\alpha y), \quad (\text{C-5})$$

i

$$(\alpha + \beta)x = (\alpha x) + (\beta x), \quad (\text{C-6})$$

– mnożenie przez skalar spełnia dodatkowo własności:

$$(\alpha\beta)x = \alpha(\beta x), \quad (\text{C-7})$$

$$\forall_{x \in X} 0x = 0, \quad (\text{C-8})$$

$$\forall_{x \in X} 1x = x. \quad (\text{C-9})$$

Używaliśmy tych samych oznaczeń na dodawanie wektorów i skalarów, należy jednak pamiętać, że $\alpha + \beta: \alpha, \beta \in \mathbb{R}$ oznacza inne działanie niż $x + y: x, y \in X$, zaś $0 \in \mathbb{R}$ i $0 \in X$ to dwa różne „zera”.

Przestrzeniami liniowymi są m.in.: n -wymiarowa przestrzeń wektorowa \mathbb{R}^n , przestrzeń ciągów liczb rzeczywistych $(a_n)_{n=0}^{\infty}$, przestrzeń funkcji ciągłych na odcinku $[a, b]$. Przestrzenią liniową *nie jest* np. zbiór ciągów nieujemnych.

Ćwiczenie C.1 Pokaż, że zbiór ciągów o skończonej liczbie wyrazów różnych od zera jest przestrzenią liniową.

Ćwiczenie C.2 Pokaż, że zbiór ciągów ograniczonych (takich, że istnieje pewne M , że $\forall_n |a_n| \leq M$) jest przestrzenią liniową.

Ćwiczenie C.3 Pokaż, że zbiór ciągów takich, że dla każdego n $|a_n| < M$ i M jest pewną wspólną dla wszystkich ciągów liczbą *nie jest* przestrzenią liniową.

C.1.2 Przestrzenie unormowane

Norma jest uogólnieniem pojęcia wartości bezwzględnej i długości wektora w \mathbb{R}^n .

Definicja C.2 Niech X będzie pewną przestrzenią liniową. Funkcję z X w \mathbb{R} przyporządkowującą wektorowi x liczbę $\|x\|$ nazywamy normą jeśli spełnia następujące warunki:

$$\|x + y\| \leq \|x\| + \|y\|, \quad (\text{C-10})$$

$$\|\alpha x\| = |\alpha| \|x\|, \quad (\text{C-11})$$

$$\forall_x \|x\| \geq 0, \quad (\text{C-12})$$

$$\|x\| = 0 \iff x = 0. \quad (\text{C-13})$$

Nierówność (C-10) nazywana jest nierównością trójkąta.

Definicja C.3 Parę $(X, \|\cdot\|)$ spełniającą warunki (C-1)-(C-13) nazywamy przestrzenią unormowaną.

Ćwiczenie C.4 Pokaż, że przestrzeń \mathbb{R}^n z normą określoną wzorem:

$$\|x\|_2 = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}$$

jest przestrzenią unormowaną.

Ćwiczenie C.5 Pokaż, że przestrzeń \mathbb{R}^n z normą określoną wzorem:

$$\|x\|_1 = |x_1| + |x_2| + \cdots + |x_n|$$

jest przestrzenią unormowaną.

Ważnym przykładem przestrzeni unormowanej jest przestrzeń ciągów ograniczonych z tzw. normą supremum określoną następująco (a jest pewnym ciągiem):

$$\|a\|_\infty \equiv \sup_n \{|a_n|\}. \quad (\text{C-14})$$

Pokażemy, że norma supremum rzeczywiście spełnia warunki (C-10)-(C-13). Spełnianie warunków (C-12) i (C-13) jest jasne. Niech α będzie pewną liczbą, zaś $(a_n)_{n=0}^\infty$ pewnym ciągiem. Wtedy:

$$\|\alpha a\| = \sup_n \{|\alpha a_n|\} = \sup_n \{|\alpha| |a_n|\} = |\alpha| \sup_n \{|a_n|\} = |\alpha| \|a\|.$$

Przedostatnia równość wynika z odpowiednika równości (1.2) dla supremum. Nierówność trójkąta pokazuje się równie łatwo (korzystamy z nierówności trójkąta dla wartości bezwzględnej i odpowiedników nierówności (1.8) i (1.6) dla supremum):

$$\|a + b\| = \sup_n \{|a_n + b_n|\} \leq \sup_n \{|a_n| + |b_n|\} \leq \sup_n \{|a_n|\} + \sup_n \{|b_n|\} = \|a\| + \|b\|.$$

Przestrzeń ciągów ograniczonych z normą zadaną przez (C-14) nazywa się przestrzenią l^∞ .

C.1.3 Metryka i zbieżność

Metryka jest naturalnym uogólnieniem pojęcia odległości punktów w \mathbb{R}^n .

Definicja C.4 Niech X będzie pewnym zbiorem. Funkcję $d: X \times X \rightarrow \mathbb{R}$ nazywamy metryką jeśli spełnia następujące warunki:

$$d(x, z) \leq d(x, y) + d(y, z), \quad (\text{C-15})$$

$$d(x, y) = d(y, x), \quad (\text{C-16})$$

$$d(x, y) \geq 0 \quad (\text{C-17})$$

$$d(x, y) = 0 \iff x = y. \quad (\text{C-18})$$

Definicja C.5 Parę $(X, d(\cdot, \cdot))$ nazywa się przestrzenią metryczną.

Dodatek C. Podstawy analizy funkcjonalnej

Podobnie jak w przypadku normy (patrz ćwiczenia C.4 i C.5) na tym samym zbiorze można zdefiniować wiele metryk. Podstawowe znaczenie metryki w matematyce sprowadza się do wyznaczanej przez nią *zbieżności*. Niech x_n będzie pewnym ciągiem elementów X i g pewnym wyróżnionym elementem X . Jeśli

$$\lim_{n \rightarrow \infty} d(x_n, g) = 0,$$

to interpretujemy to jako wyraz tego, że elementy ciągu x_n są *coraz bliżej* (zbiegają do) g . Rozwijając definicję granicy ciągu liczbowego możemy poprzez metrykę zdefiniować granicę ciągu w przestrzeni metrycznej.

Definicja C.6 Niech x_n będzie ciągiem elementów przestrzeni $(X, d(\cdot, \cdot))$ i g pewnym jej elementem. Mówimy, że g jest granicą ciągu x_n w metryce $d(\cdot, \cdot)$ jeśli:

$$\forall \varepsilon > 0 \exists N \forall n > N \quad d(x_n, g) < \varepsilon. \quad (\text{C-19})$$

Zbieżność ciągu do granicy g oznacza, że odpowiednio dalekie jego wyrazy są dowolnie blisko g . Podamy jeszcze jedną definicję.

Definicja C.7 Kulą otwartą o promieniu r i środku x nazywamy zbiór:

$$K(x, r) = \{y \in X : d(x, y) < r\}. \quad (\text{C-20})$$

Ćwiczenie C.6 Narysuj „kulę” otwartą $K(0, 1)$ w przestrzeni \mathbb{R}^2 z metryką:

$$\begin{aligned} d_1(x, y) &= |x_1 - y_1| + |x_2 - y_2|, \\ d_2(x, y) &= \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2}, \\ d_\infty(x, y) &= \max\{|x_1 - y_1|, |x_2 - y_2|\}. \end{aligned}$$

Korzystając z pojęcia kuli nierówność (C-19) można zinterpretować następująco: od pewnego N wyrazy ciągu x_n leżą w arbitralnie małej kuli otwartej o środku g . Zbiory otwarte definiuje się jako zbiory, których każdy punkt jest środkiem pewnej kuli otwartej zawierającej się w tym zbiorze. Zbiór otwarty zawierający punkt x nazywamy otoczeniem tego punktu. Zauważmy, że definicję granicy można uogólnić następująco: x_n ma granicę g jeśli od pewnego N wyrazy x_n leżą w arbitralnym otoczeniu g . Taką ogólną definicją granicy posługuje się dział matematyki zwany topologią.

Uważny Czytelnik z pewnością zauważył, że norma w naturalny sposób zadaje metrykę. Dowód tego faktu pozostawiamy jako ćwiczenie.

Fakt C.1 Przestrzeń unormowana $(X, \|\cdot\|)$ jest przestrzenią metryczną z metryką generowaną przez normę:

$$d(x, y) = \|x - y\| \quad (\text{C-21})$$

Ćwiczenie C.7 Udowodnij powyższy fakt sprawdzając warunki (C-15)-(C-18). **Wskazówka:** $x - z = x - y + y - z$.

Ćwiczenie C.8 Niech $(a_n)_{n=0}^\infty$ będzie ciągiem stałym $a_n = 1$. Czym jest kula $K(a; 1)$ w przestrzeni l^∞ ? Czy ciąg $a_n = 1/n$ należy do $K(a; 1)$?

Należy się jeszcze jedna uwaga. W przestrzeniach \mathbb{R}^n wszystkie normy (m.in. $\|\cdot\|_1, \|\cdot\|_2, \|\cdot\|_\infty$) zadają taką samą zbieżność (taką samą topologię), to znaczy, jeśli x_n zbiega do g w jednej z norm zbiega też we wszystkich innych. W nieskończeniowych przestrzeniach tak nie jest.

C.1.4 Ciągi Cauchy'ego i zupełność

Fundamentalnym zagadnieniem pojawiającym się w wielu zastosowaniach jest określenie czy pewien ciąg x_n ma granicę w X . x_n może np. być ciągiem funkcji uzyskanych metoda kolejnych przybliżeń, szeregiem funkcyjnym czy też ciągiem ciągów.

Rozpocznijmy od definicji.

Definicja C.8 Ciąg x_n nazywamy ciągiem Cauchy'ego jeśli:

$$\forall \varepsilon > 0 \exists N \forall n, m > N \quad d(x_n, x_m) < \varepsilon. \quad (\text{C-22})$$

Wszystkie odpowiednio dalekie wyrazy ciągu Cauchy'ego są więc dowolnie blisko siebie.

Fakt C.2 Niech ciąg x_n ma granicę g . Wtedy x_n jest ciągiem Cauchy'ego.

Dowód. Ustalmy $\varepsilon > 0$. Z definicji granicy istnieje takie N , że dla $n > N$ każde x_n spełnia warunek:

$$d(x_n, g) < \frac{\varepsilon}{2}.$$

Niech n i m będą dowolnymi indeksami większymi niż N . Z nierówności trójkąta (C-15) mamy:

$$d(x_n, x_m) \leq d(x_n, g) + d(x_m, g) < \varepsilon.$$

Ponieważ liczbę ε wzięliśmy dowolną, więc ciąg x_n spełnia warunki definicji C.8. ■

Będą nas interesowały przestrzenie, w których zachodzi twierdzenie odwrotne, a więc ciągi Cauchy'ego mają granicę.

Definicja C.9 Przestrzeń metryczną $(X, d(\cdot, \cdot))$ nazywamy przestrzenią zupełną, jeśli każdy ciąg Cauchy'ego ma w niej granicę.

Przestrzenią zupełną *nie jest* np. przestrzeń liczb wymiernych \mathbb{Q} z metryką $d(x, y) = |x - y|$. By to dostrzec wystarczy wziąć dowolny ciąg liczb wymiernych zbiegający do liczby niewymiernej (np $a_n = (1 + 1/n)^n$).

Zachodzi następujące twierdzenie, które podajemy bez dowodu.

Twierdzenie C.1 Przestrzeń liczb rzeczywistych \mathbb{R} z metryką zadaną przez wartość bezwzględną jest zupełna.

Podamy teraz podstawową definicję.

Definicja C.10 Przestrzeń unormowaną $(X, \|\cdot\|)$ nazywamy przestrzenią Banacha jeśli jest zupełna (w metryce generowanej przez normę).

Standardową procedurą przy rozpatrywaniu zagadnień istnienia granicy pewnego ciągu będzie zapewnienie, iż jest on ciągiem elementów przestrzeni Banacha i pokazanie, że ma własność Cauchy'ego. Przed pokazaniem przykładów zastosowania tej metody udowodnimy, że dwie bardzo ważne przestrzenie są przestrzeniami Banacha.

 Dodatek C. Podstawy analizy funkcjonalnej

Twierdzenie C.2 *Przestrzeń l^∞ jest przestrzenią Banacha.*

Dowód. Weźmy ciąg Cauchy'ego ciągów a^n . Z definicji mamy:

$$\forall \varepsilon > 0 \exists N \forall n, m > N \sup_i \{|a_i^m - a_i^n|\} < \varepsilon.$$

Wynika stąd wprost, że:

$$\forall_i \forall \varepsilon > 0 \exists N \forall n, m > N |a_i^n - a_i^m| < \varepsilon,$$

a więc ciągi liczb a_i^n (i ustalone) są ciągami Cauchy'ego. Z zupełności przestrzeni liczb rzeczywistych \mathbb{R} wynika, że każdy taki ciąg ma granicę. Oznaczmy:

$$a_i = \lim_{n \rightarrow \infty} a_i^n.$$

Pokażemy, że (a) $\lim_{n \rightarrow \infty} \|a - a^n\|_\infty = 0$ i (b) a jest ciągiem ograniczonym.

(a) Ustalmy $\varepsilon > 0$. Z własności Cauchy'ego ciągu a^n możemy wybrać taką liczbę naturalną N , że dla każdego $n, m > N$ $\|a^n - a^m\| < \varepsilon/2$. Ustalmy teraz pewne i . Mamy:

$$\lim_{m \rightarrow \infty} |a_i^n - a_i^m| = |a_i^n - a_i|.$$

Ciąg liczb mniejszych niż $\varepsilon/2$ ma granicę nie większą od $\varepsilon/2$. Ponieważ indeks i był wybrany arbitralnie mamy:

$$\forall \varepsilon > 0 \exists N \forall n > N \forall_i |a_i^n - a_i| \leq \varepsilon/2,$$

skąd wynika, że:

$$\forall \varepsilon > 0 \exists N \forall n > N \|a - a^n\|_\infty = \sup_i \{|a_i^n - a_i|\} \leq \varepsilon/2 < \varepsilon.$$

(b) Pokażemy, że istnieje taka liczba rzeczywista M' , że $\sup_i \{|a_i|\} \leq M'$. Ustalmy liczbę $\varepsilon > 0$ i weźmy taką liczbę naturalną n , by było $\|a - a^n\|_\infty < \varepsilon$. Ponieważ a^n jest ciągiem ograniczonym, istnieje taka liczba rzeczywista M , że $\sup_i \{|a_i^n|\} \leq M$. Mamy wówczas:

$$\sup_i \{|a_i|\} \leq \sup_i \{|a_i - a_i^n| + |a_i^n|\} \leq \sup_i \{|a_i - a_i^n|\} + \sup_i \{|a_i^n|\} \leq \varepsilon + M. \quad \blacksquare$$

Ćwiczenie C.9 Pokaż, że przestrzeń wektorowa \mathbb{R}^n z każdą z norm:

$$\|x\|_1 = \sum_{i=1}^n |x_i|,$$

$$\|x\|_2 = \sqrt{\sum_{i=1}^n x_i^2},$$

$$\|x\|_\infty = \sup_i \{|x_i|\} (= \max_i \{|x_i|\})$$

jest przestrzenią Banacha.

Uwaga: w ogólności (nieskończeniowymymiarowych przestrzeniach) to, czy dana przestrzeń jest lub nie przestrzenią Banacha zależy istotnie od obranej normy.

Dowód poniższego twierdzenia jest bardzo podobny do poprzedniego.

Twierdzenie C.3 Przestrzeń $B(X)$ funkcji ciągłych i ograniczonych na $X \subset \mathbb{R}^1$ w \mathbb{R} z normą supremum²

$$\|f\| = \sup_{x \in X} \{|f(x)|\}$$

jest przestrzenią Banacha.

Dowód. Niech f_n będzie ciągiem Cauchy'ego funkcji z $B(X)$. Z definicji mamy:

$$\forall \varepsilon > 0 \exists N \forall n, m > N \sup_{x \in X} \{|f_n(x) - f_m(x)|\} < \varepsilon.$$

Wynika stąd wprost, że:

$$\forall x \forall \varepsilon > 0 \exists N \forall n, m > N |f_n(x) - f_m(x)| < \varepsilon,$$

a więc ciągi liczb $f_n(x)$ są ciągami Cauchy'ego. Z zupełności przestrzeni liczb rzeczywistych \mathbb{R} wynika, że każdy taki ciąg ma granicę. Oznaczmy:

$$f(x) = \lim_{n \rightarrow \infty} f_n(x).$$

Pokażemy, że (a) $\lim_{n \rightarrow \infty} \|f - f_n\| = 0$ i (b) f jest ciągła na X . To, że f jest ograniczona wynika wprost z (a), podobnie jak w poprzednim dowodzie.

(a) Ustalmy liczbę $\varepsilon > 0$. Z własności Cauchy'ego ciągu f_n możemy wybrać taką liczbę naturalną N , że dla każdego $n, m > N$ $\|f_n - f_m\| < \varepsilon/2$. Ustalmy teraz element $x \in X$. Mamy:

$$\lim_{m \rightarrow \infty} |f_n(x) - f_m(x)| = |f_n(x) - f(x)|.$$

Ciąg liczb mniejszych niż $\varepsilon/2$ ma granicę nie większą od $\varepsilon/2$. Ponieważ element x był wybrany arbitralnie mamy:

$$\forall \varepsilon > 0 \exists N \forall n > N \forall x |f_n(x) - f(x)| \leq \varepsilon/2,$$

skąd wynika, że:

$$\forall \varepsilon > 0 \exists N \forall n > N \|f - f_n\| = \sup_x \{|f_n(x) - f(x)|\} \leq \varepsilon/2 < \varepsilon.$$

(b) Ciągłość oznacza, że:

$$\forall x \in X \forall \varepsilon > 0 \exists \delta > 0 \forall y \in X: |x - y| < \delta |f(x) - f(y)| < \varepsilon.$$

Ustalmy element $x \in X$ i liczbę $\varepsilon > 0$. Weźmy taką liczbę naturalną n , by $\|f - f_n\| < \varepsilon/3$. Z ciągłości funkcji f_n wynika, że możemy wziąć taką liczbę $\delta > 0$ by spełniony był warunek:

$$\forall y: |y - x| < \delta |f_n(x) - f_n(y)| < \varepsilon/3.$$

Wówczas:

$$\forall y: |y - x| < \delta |f(x) - f(y)| < \varepsilon.$$

Rzeczywiście, mamy:

$$\begin{aligned} |f(x) - f(y)| &\leq |f(x) - f_n(x)| + |f_n(x) - f_n(y)| + |f_n(y) - f(y)| \leq \\ &2 \|f_n - f\| + |f_n(x) - f_n(y)| < \frac{2}{3}\varepsilon + \frac{1}{3}\varepsilon = \varepsilon. \quad \blacksquare \end{aligned}$$

¹ Podobnie jest gdy X jest podzbiorem dowolnej przestrzeni metrycznej. Wystarczy w dowodzie zastąpić $|x - y|$ przez $d_X(x, y)$ w definicji ciągłości funkcji z X w \mathbb{R} .

² Norma supremum określa znaną z kursu analizy zbieżność jednostajną.

C.1.5 Odwzorowania zblizajace

Definicja C.11 Niech T bedzie pewnym odwzorowaniem przestrzeni metrycznej $(X, d(\cdot, \cdot))$ w siebie. Mowimy, ze T jest odwzorowaniem zblizajacym, jezeli

$$\forall_{x, y \in X} d(T(x), T(y)) \leq \beta d(x, y), \quad (\text{C-23})$$

gdzie β jest pewna liczba z przedzialu $(0, 1)$.

Odwzorowanie zblizajace „sciaga” punkty przestrzeni blizej siebie. Jezeli oznaczmy

$$T^n(x) = \underbrace{T(T(\dots T(x)))}_{n \text{ razy}},$$

to z nierownosci (C-23) uzyskamy:

$$d(T^n(x), T^n(y)) \leq \beta^n d(x, y),$$

co oznacza, ze w nieskonczoności x i y zostana sciagniete do jednego punktu (bo $\beta^n \rightarrow 0$). Nalezny oczekiwac, ze w calaj przestrzeni taki punkt przyciagajacy bedzie jeden. Powyzsze intuicje formalizuje nastepujace twierdzenie.

Twierdzenie C.4 (Banacha o odwzorowaniu zblizajacym) Niech T bedzie odwzorowaniem zblizajacym przestrzeni Banacha $(X, \|\cdot\|)$ w siebie. Wtedy istnieje jedyne takie $g \in X$, ze $\forall_{x \in X} \lim_{n \rightarrow \infty} T^n(x) = g$ i $T(g) = g$.

Dowód. Ustalmy x i zdefiniujmy ciag $x_n = T^n(x)$. Wezmy pewne liczby naturalne n i m (dla porzadku niech $m > n$). Mamy:

$$\|x_{n+1} - x_n\| \leq \beta^n \|x_1 - x\|.$$

Z nierownosci trojkata uzyskujemy :

$$\begin{aligned} \|x_m - x_n\| &\leq \|x_m - x_{m-1}\| + \|x_{m-1} - x_{m-2}\| + \dots + \|x_{n+1} - x_n\| \leq \\ &(\beta^{m-1} + \beta^{m-2} + \dots + \beta^n) \|x_1 - x\| = \\ &(\beta^{m-n-1} + \beta^{m-n-2} + \dots + 1)\beta^n \|x_1 - x\| = \\ &\frac{1 - \beta^{m-n}}{1 - \beta} \beta^n \|x_1 - x\| \leq \frac{\beta^n}{1 - \beta} \|x_1 - x\|. \end{aligned}$$

Ustalmy liczbe $\varepsilon > 0$ i wezmy taka liczbe naturalna N , by $\beta^N \|x_1 - x\| / (1 - \beta) < \varepsilon$. Wtedy dla kazdych $n, m > N$ $\|x_m - x_n\| < \varepsilon$, a wiec x_n jest ciagiem Cauchy'ego. Poniewaz x_n jest ciagiem elementow przestrzeni kompletnej, wiec ma w niej granice. Oznaczmy ja g^x . Pokazemy, ze $T(g^x) = g^x$. Ustalmy liczbe $\varepsilon > 0$. Wezmy taka liczbe naturalna n by $\|x_{n-1} - g^x\| < \varepsilon/2$. Z nierownosci trojkata i wlasnosci odwzorowan zblizajacych mamy:

$$\begin{aligned} \|T(g^x) - g^x\| &\leq \|T(g^x) - x_n\| + \|x_n - g^x\| = \|T(g^x) - T(x_{n-1})\| + \|x_n - g^x\| \leq \\ &\|g^x - x_{n-1}\| + \|x_n - g^x\| < \varepsilon. \end{aligned}$$

Ponieważ liczba ε było wybrana dowolnie, mamy $\|T(g^x) - g^x\| = 0$, a więc $T(g^x) = g^x$. Pokażemy, że T ma tylko jeden punkt stały. Niech $g \neq g'$ i $T(g) = g$ oraz $T(g') = g'$. Z założeń tych wynika:

$$\|g - g'\| = \|T(g) - T(g')\| \leq \beta \|g - g'\|.$$

Ponieważ $\|g - g'\| > 0$ (bo $g \neq g'$) i jednocześnie $\beta < 1$ uzyskujemy sprzeczność. ■

Rozpatrzmy problem następującego równania:

$$T(x) = x. \quad (\text{C-24})$$

Jeśli poszukujemy x w pewnej przestrzeni Banacha i umiemy pokazać, że T jest odwzorowaniem zwężającym, wtedy twierdzenie Banacha gwarantuje nam, że równanie (C-24) ma rozwiązanie jednoznaczne i można je uzyskać metodą kolejnych przybliżeń starując od pewnego x_0 i definiując ciąg x_n rekurencyjnie przez $x_n = T(x_{n-1})$.

Zauważmy, że równanie Bellmana w problemie z nieskończonym horyzontem (patrz rozdział 3) ma właśnie postać (C-24), gdzie operator T jest określony wzorem:

$$T(f)(s_t) = \sup_{s_{t+1}} \{u(s_t, s_{t+1}) + \beta f(s_{t+1})\}, \quad (\text{C-25})$$

zaś f (funkcja wartości) jest elementem pewnej przestrzeni funkcyjnej. W przypadku, gdy poszukujemy funkcji wartości w przestrzeni funkcji $B(S)$ ciągłych ograniczonych z pewnego podzbioru $S \subset \mathbb{R}^n$ w \mathbb{R} z normą supremum, poniżej prezentowane twierdzenie Blackwella zapewnia, że T jest odwzorowaniem zbliżającym. Oczywiście by T było odwzorowaniem z $B(S)$ w siebie, funkcja użyteczności chwilowej $u(s_t, s_{t+1})$ musi być ograniczona a tak w ogólnym przypadku nie jest.

Twierdzenie C.5 (Blackwella) Niech $B(S)$ będzie określone jak wyżej, zaś odwzorowanie $T: B(S) \rightarrow B(S)$ spełnia warunki:

$$f(x) \leq g(x): f, g \in B(S) \Rightarrow T(f)(x) \leq T(g)(x) \quad (\text{C-26})$$

i

$$T(f + a)(x) \leq T(f)(x) + \beta a, \quad (\text{C-27})$$

przy czym $0 < \beta < 1$ i $(f + a)(x)$ oznacza $f(x) + a$. Wtedy T jest odwzorowaniem zwężającym ze współczynnikiem β .

Dowód. Weźmy dowolne funkcje $f, g \in B(S)$. Z definicji normy na $B(S)$ wynika, że $f \leq g + \|f - g\|$. Z założeń o T mamy:

$$T(f) \leq T(g + \|f - g\|) \leq T(g) + \beta \|f - g\|,$$

skąd otrzymujemy:

$$T(f) - T(g) \leq \beta \|f - g\|.$$

Jednocześnie wiemy, że $g \leq f + \|f - g\|$ i wobec tego:

$$T(g) \leq T(f + \|f - g\|) \leq T(f) + \beta \|f - g\|,$$

a stąd:

$$-\beta \|f - g\| \leq T(f) - T(g).$$

Z faktu, że $-\beta \|f - g\| \leq T(f) - T(g) \leq \beta \|f - g\|$ wynika, że $\|T(f) - T(g)\| \leq \beta \|f - g\|$ a więc T jest odwzorowaniem zwężającym. ■

Ćwiczenie C.10 Pokaż, że operator (C-25) spełnia nierówności (C-26) i (C-27).

Szczegółów teorii programowania dynamicznego z ograniczoną funkcją wypłaty chwilowej nie podajemy odsyłając zainteresowanego Czytelnika do monografii Stockey, Lucasa i Prescottta [13].

C.1.6 Przestrzenie l^p

Przestrzenie l^p są ważną klasą przestrzeni Banacha i pojawią się później w dyskusji funkcjonałów liniowych. Przed podaniem ich definicji udowodnimy dwie bezpośrednio związane z nimi nierówności.

W dowodzie nierówności Höldera wykorzystamy prosty lemat. Zakładamy, że Czytelnikowi znane jest pojęcie wypukłości funkcji $f: \mathbb{R} \rightarrow \mathbb{R}$.

Lemat C.1 Niech $0 < \lambda < 1$ i $x, y \geq 0$. Wtedy:

$$x^\lambda y^{1-\lambda} \leq \lambda x + (1-\lambda)y. \quad (\text{C-28})$$

Dowód. Jeśli $x = 0$ lub $y = 0$ nierówność jest oczywista. Niech więc będzie $x, y > 0$. Z wypukłości funkcji wykładniczej mamy:

$$x^\lambda y^{1-\lambda} = e^{\lambda \ln x + (1-\lambda) \ln y} \leq \lambda e^{\ln x} + (1-\lambda)e^{\ln y} = \lambda x + (1-\lambda)y. \quad \blacksquare$$

Twierdzenie C.6 (Nierówność Höldera) Niech liczby $1 < p, q < \infty$ (są to tzw. wykładniki sprzężone) spełniają warunek:

$$\frac{1}{p} + \frac{1}{q} = 1. \quad (\text{C-29})$$

Niech a_n i b_n będą ciągami, dla których:

$$\sum_{n=1}^{\infty} |a_n|^p < \infty \quad (\text{C-30})$$

i

$$\sum_{n=1}^{\infty} |b_n|^q < \infty. \quad (\text{C-31})$$

Wtedy:

$$\sum_{n=1}^{\infty} |a_n b_n| \leq \left(\sum_{n=1}^{\infty} |a_n|^p \right)^{\frac{1}{p}} \left(\sum_{n=1}^{\infty} |b_n|^q \right)^{\frac{1}{q}}. \quad (\text{C-32})$$

Dowód. W przypadku, gdy $\sum_{n=1}^{\infty} |a_n|^p = 0$ lub $\sum_{n=1}^{\infty} |b_n|^q = 0$ spełnianie warunku (C-32) jest jasne, bo mamy albo $\forall_n a_n = 0$, albo $\forall_n b_n = 0$, co implikuje $\sum_{n=1}^{\infty} a_n b_n = 0$.

Założmy więc, że $A \equiv (\sum_{n=1}^{\infty} |a_n|^p)^{1/p} > 0$ i $B \equiv (\sum_{n=1}^{\infty} |b_n|^q)^{1/q} > 0$. Niech n będzie dowolną liczbą naturalną. Przyjmijmy w lemacie C.1 $\lambda = 1/p$ ($1 - \lambda = 1/q$), $x = (|a_n|/A)^p$ i $y = (|b_n|/B)^q$. Mamy:

$$\left[\left(\frac{|a_n|}{A} \right)^p \right]^{\frac{1}{p}} \left[\left(\frac{|b_n|}{B} \right)^q \right]^{\frac{1}{q}} \leq \frac{1}{p} \frac{|a_n|^p}{A^p} + \frac{1}{q} \frac{|b_n|^q}{B^q},$$

skąd wynika, że dla każdego n :

$$\frac{1}{AB} |a_n b_n| \leq \frac{1}{pA^p} |a_n|^p + \frac{1}{qB^q} |b_n|^q.$$

Biorąc sumy ciągów po obu stronach nierówności uzyskujemy:

$$\frac{1}{AB} \sum_{n=1}^{\infty} |a_n b_n| \leq \frac{1}{pA^p} \sum_{n=1}^{\infty} |a_n|^p + \frac{1}{qB^q} \sum_{n=1}^{\infty} |b_n|^q.$$

Z określenia A i B mamy $\sum_{n=1}^{\infty} |a_n|^p = A^p$ i $\sum_{n=1}^{\infty} |b_n|^q = B^q$, co daje:

$$\frac{1}{AB} \sum_{n=1}^{\infty} |a_n b_n| \leq \frac{1}{p} + \frac{1}{q} = 1.$$

Ostatecznie:

$$\sum_{n=1}^{\infty} |a_n b_n| \leq AB. \quad \blacksquare$$

Jest jasne, że powyższe twierdzenie zachodzi dla sum skończonych tzn. dla dowolnych ciągów $(a_n)_{n=1}^N$ i $(b_n)_{n=1}^N$ mamy:

$$\sum_{n=1}^N |a_n b_n| \leq \left(\sum_{n=1}^N |a_n|^p \right)^{\frac{1}{p}} \left(\sum_{n=1}^N |b_n|^q \right)^{\frac{1}{q}}. \quad (\text{C-33})$$

Szczególny przypadek nierówności Höldera, gdy $p = q = 2$ nazywany jest nierównością Schwarzera.

Korzystając z nierówności Höldera wyprowadzimy teraz nierówność Minkowskiego.

Twierdzenie C.7 (Nierówność Minkowskiego) Niech $1 \leq p < \infty$, zaś a_n i b_n będą ciągami, dla których:

$$\sum_{n=1}^{\infty} |a_n|^p < \infty \quad (\text{C-34})$$

i

$$\sum_{n=1}^{\infty} |b_n|^p < \infty. \quad (\text{C-35})$$

Wtedy:

$$\left(\sum_{n=1}^{\infty} |a_n + b_n|^p \right)^{\frac{1}{p}} \leq \left(\sum_{n=1}^{\infty} |a_n|^p \right)^{\frac{1}{p}} + \left(\sum_{n=1}^{\infty} |b_n|^p \right)^{\frac{1}{p}}. \quad (\text{C-36})$$

Dodatek C. Podstawy analizy funkcjonalnej

Dowód. W przypadku, gdy $\sum_{n=1}^{\infty} |a_n|^p = 0$ lub $\sum_{n=1}^{\infty} |b_n|^p = 0$ nierówność jest jasna, bo mamy albo $\forall_n a_n = 0$, albo $\forall_n b_n = 0$, i warunek (C-36) redukuje się do nierówności $(\sum_{n=1}^{\infty} |b_n|^p)^{\frac{1}{p}} \leq (\sum_{n=1}^{\infty} |b_n|^p)^{\frac{1}{p}}$ lub $(\sum_{n=1}^{\infty} |a_n|^p)^{\frac{1}{p}} \leq (\sum_{n=1}^{\infty} |a_n|^p)^{\frac{1}{p}}$. Niech $p = 1$. Wtedy z nierówności trójkąta mamy dla każdego n :

$$|a_n + b_n| \leq |a_n| + |b_n|,$$

co po zsumowaniu daje:

$$\sum_{n=1}^{\infty} |a_n + b_n| \leq \sum_{n=1}^{\infty} |a_n| + \sum_{n=1}^{\infty} |b_n|.$$

Niech więc $1 < p < \infty$, $\sum_{n=1}^{\infty} |a_n|^p > 0$ i $\sum_{n=1}^{\infty} |b_n|^p > 0$. Z wypukłości funkcji x^p mamy:

$$\left(\frac{1}{2}|a_n| + \frac{1}{2}|b_n|\right)^p \leq \frac{1}{2}|a_n|^p + \frac{1}{2}|b_n|^p,$$

czyli:

$$(|a_n| + |b_n|)^p \leq 2^{p-1}(|a_n|^p + |b_n|^p).$$

Jednocześnie z tego, że $|a_n + b_n| \leq |a_n| + |b_n|$ wynikają nierówności:

$$|a_n + b_n|^p \leq (|a_n| + |b_n|)^p \leq 2^{p-1}(|a_n|^p + |b_n|^p).$$

Biorąc sumy po każdym z wyrazów tej nierówności stwierdzamy, że jeśli prawa strona (C-36) jest skończona, to skończona jest także lewa strona i skończona jest także suma $\sum_{n=1}^{\infty} (|a_n| + |b_n|)^p$.

Zauważmy, że

$$(|a_n| + |b_n|)^p = |a_n|(|a_n| + |b_n|)^{p-1} + |b_n|(|a_n| + |b_n|)^{p-1}.$$

Niech q będzie wykładnikiem sprzężonym z p . Wtedy zachodzi $(p-1)q = p$. Z nierówności Hödera mamy:

$$\sum_{n=1}^{\infty} |a_n| (|a_n| + |b_n|)^{p-1} \leq \left(\sum_{n=1}^{\infty} |a_n|^p\right)^{\frac{1}{p}} \left(\sum_{n=1}^{\infty} (|a_n| + |b_n|)^{(p-1)q}\right)^{\frac{1}{q}}.$$

i

$$\sum_{n=1}^{\infty} |b_n| (|a_n| + |b_n|)^{p-1} \leq \left(\sum_{n=1}^{\infty} |b_n|^p\right)^{\frac{1}{p}} \left(\sum_{n=1}^{\infty} (|a_n| + |b_n|)^{(p-1)q}\right)^{\frac{1}{q}}.$$

Dodając powyższe nierówności stronami otrzymujemy:

$$\sum_{n=1}^{\infty} (|a_n| + |b_n|)^p \leq \left(\sum_{n=1}^{\infty} (|a_n| + |b_n|)^p\right)^{\frac{1}{q}} \left[\left(\sum_{n=1}^{\infty} |a_n|^p\right)^{\frac{1}{p}} + \left(\sum_{n=1}^{\infty} |b_n|^p\right)^{\frac{1}{p}} \right],$$

skąd wynika nierówność:

$$\left(\sum_{n=1}^{\infty} (|a_n| + |b_n|)^p\right)^{\frac{1}{p}} \leq \left(\sum_{n=1}^{\infty} |a_n|^p\right)^{\frac{1}{p}} + \left(\sum_{n=1}^{\infty} |b_n|^p\right)^{\frac{1}{p}}.$$

Oczywiście

$$\left(\sum_{n=1}^{\infty} |a_n + b_n|^p\right)^{\frac{1}{p}} \leq \left(\sum_{n=1}^{\infty} (|a_n| + |b_n|)^p\right)^{\frac{1}{p}},$$

do daje tezę. ■

Mając udowodnione powyższe nierówności, możemy przystąpić do definicji przestrzeni l^p .

Definicja C.12 Przestrzemią l^p dla $1 \leq p < \infty$ nazywamy przestrzeń ciągów takich, że dla każdego $(a_n)_{n=1}^{\infty} \in l^p$ suma:

$$\sum_{n=1}^{\infty} |a_n|^p$$

jest skończona. Normę w l^p określamy wzorem:

$$\|a\|_p = \left(\sum_{n=1}^{\infty} |a_n|^p \right)^{\frac{1}{p}}. \quad (\text{C-37})$$

Przestrzeń l^{∞} to określona już wcześniej przestrzeń ciągów ograniczonych z normą (C-14).

To, że $\|\cdot\|_{\infty}$ jest normą już wykazywaliśmy. Spełnianie warunków (C-11)-(C-13) dla $1 \leq p < \infty$ jest jasne. Z nierówności Minkowskiego wynika wprost fakt, iż $\|\cdot\|_p$ spełnia nierówność trójkąta.

Udowodniliśmy już, że l^{∞} jest przestrzenią Banacha.

Twierdzenie C.8 Przestrzenie l^p dla $1 \leq p < \infty$ są przestrzeniami Banacha.

Ćwiczenie C.11 Udowodnij twierdzenie C.8 wzorując się na dowodzie twierdzenia C.2, tzn. pokaż, że jeśli a_i^n ciągiem Cauchy'ego w $\|\cdot\|_p$ to dla ustalonego i a_i^n jest ciągiem Cauchy'ego w \mathbb{R} ; pokaż dalej, że ciąg $a_i = \lim_{n \rightarrow \infty} a_i^n$ należy l^p i a_i jest granicą w $\|\cdot\|_p$ ciągu a_i^n .

Pokażemy teraz jak bardzo przestrzenie l^p różnią się od skończeniowymiarowych przestrzeni \mathbb{R}^n . Zaczniemy od formalnej definicji zbioru otwartego.

Definicja C.13 Zbiorem otwartym w przestrzeni metrycznej $(X, d(\cdot, \cdot))$ nazywamy każdy taki zbiór $A \subset X$, że dla każdego $x \in A$ istnieje kula otwarta (patrz definicja C.7) o środku w x i pewnym dodatnim promieniu $r > 0$, należąca do A .

Innymi słowy, zbiór otwarty nie ma brzegu, każdy jego punkt otoczony jest innymi punktami tego zbioru.

Definicja C.14 Zbiorem domkniętym w przestrzeni metrycznej $(X, d(\cdot, \cdot))$ nazywamy każdy taki zbiór $A \subset X$, którego dopełnienie $X \setminus A$ jest zbiorem otwartym.

Fakt C.3 Zbiór $A = \mathbb{R}_+^n \subset \mathbb{R}^n$ jest zbiorem otwartym w metryce zadanej przez $\|\cdot\|_p$.

Dowód. Niech $x \in A$. Weźmy $2r = \min_n \{x_n\} > 0$. Kula $K(x, r)$ leży w A .

W przestrzeni l^p zbiór ciągów, takich, że $\forall_n a_n > 0$ nie jest zbiorem otwartym. Weźmy dla przykładu ciąg $a_n = 1/n^{\alpha}$, gdzie $\alpha > 1$ (patrz też ćwiczenie C.8). Ciąg ten należy do l^p , dla każdego $1 \leq p \leq \infty$. Pokażemy, że w każdej kuli o środku w a leży ciąg, którego jeden wyraz jest niedodatni. Ustalmy $\varepsilon > 0$. Weźmy takie N , by było $N > (2/\varepsilon)^{1/\alpha}$. Niech a'_n , będzie ciągiem takim, że $a'_n = a_n : n \neq N$ i $a'_N = a_N - \varepsilon/2 < 0$. Jednocześnie a' leży w kuli o środku a i promieniu ε , bo:

$$\|a - a'\|_p = \frac{\varepsilon}{2}.$$

Definicja C.15 Wnętrzem zbioru $A \subset X$ nazywamy największy podzbiór zbioru A otwarty w X . Wnętrze zbioru A oznaczamy przez $\text{int } A$.

W przestrzeniach l^p dla $1 \leq p < \infty$ zbiór ciągów dodatnich ma *puste* wnętrze, tzn. każdy taki ciąg należący do l^p ($1 \leq p < \infty$) jest otoczony przez ciągi, których mają wyrazy niedodatnie. Wystarczy zauważyć, że każdy ciąg z l^p ($1 \leq p < \infty$) musi zbiegać do zera, a więc możemy zastosować wcześniejszy argument (odjąć od odpowiednio małego wyrazu $\varepsilon/2$), by pokazać, że w każdym epsilonowym otoczeniu każdego (dodatniego) ciągu z l^p ($1 \leq p < \infty$) leży ciąg o co najmniej jednym wyrazie niedodatnim.

C.2 Odwzorowania liniowe

C.2.1 Odwzorowania liniowe ciągłe

Definicja C.16 Odwzorowanie A przestrzeni Banacha $(X, \|\cdot\|_X)$ w przestrzeń Banacha $(Y, \|\cdot\|_Y)$ nazywamy liniowym, jeśli:

$$\forall \alpha, \beta \in \mathbb{R} \forall x_1, x_2 \in X \quad A(\alpha x_1 + \beta x_2) = \alpha A(x_1) + \beta A(x_2). \quad (\text{C-38})$$

Będziemy pisali Ax , zamiast $A(x)$. Oczywiście działania liniowe można dodawać do siebie i mnożyć przez skalary uzyskując w wyniku znów działania liniowe, a więc odwzorowania liniowe mają naturalną strukturę przestrzeni liniowej.

Niech $X = \mathbb{R}^m$ i $Y = \mathbb{R}^n$, wtedy każde odwzorowanie liniowe $A: X \rightarrow Y$ można utożsamić z macierzą $(A_{i,j})$ (o wymiarze $n \times m$) i pisać

$$A(x) = (A_{i,j})x,$$

gdzie po prawej stronie mamy zwykle mnożenie macierzy przez wektor. Niech $x \in X$. Wtedy można zapisać:

$$x = \sum_{j=1}^m x_j e_j,$$

gdzie e_i są wersorami (wektorami, których i -ta współrzędna jest równa 1 a pozostałe 0). Mamy (z liniowości A):

$$Ax = \sum_{j=1}^m x_j A e_j = \begin{pmatrix} A e_1 & \cdots & A e_m \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix}.$$

Z powyższego wynika wprost, że j -ta kolumna $(A_{i,j})$ to po prostu wektor $A e_j$.

Definicja C.17 Odwzorowanie liniowe $A: X \rightarrow Y$ nazywamy ograniczonym jeśli istnieje takie C , że

$$\forall x \in X \quad \|Ax\|_Y \leq C \|x\|_X. \quad (\text{C-39})$$

Najmniejszą liczbę C , która może stać po prawej stronie powyższej nierówności określimy jako normę A :

$$\|A\| \equiv \sup_x \left\{ \frac{\|Ax\|_Y}{\|x\|_X} \right\} = \sup_x \left\{ \left\| A \frac{x}{\|x\|_X} \right\|_Y \right\} = \sup_{x:\|x\|=1} \{\|Ax\|_Y\}. \quad (\text{C-40})$$

Ostatnia równość wynika, z tego, że:

$$\left\| \frac{x}{\|x\|_X} \right\|_X = \frac{\|x\|_X}{\|x\|_X} = 1.$$

Ćwiczenie C.12 Pokaż, że $\|A\|$ określone przez (C-40) spełnia warunki podane w definicji normy.

Ćwiczenie C.13 Niech $X = \mathbb{R}^m$ i $Y = \mathbb{R}^n$. Określ normę $\|A\|$ odwzorowania $A: X \rightarrow Y$ jako funkcję wyrazów macierzy $(A_{i,j})$, gdy:

- na X określona jest norma $\|\cdot\|_1$, na Y – norma $\|\cdot\|_\infty$,
- na X określona jest norma $\|\cdot\|_\infty$, na Y – norma $\|\cdot\|_1$.

W skończeniowym wymiarach przestrzeniach każde odwzorowanie liniowe jest ciągłe i ograniczone. W ogólności tak nie jest. Własności te są jednak ze sobą mocno powiązane. Warunkiem koniecznym i dostatecznym ciągłości odwzorowania liniowego jest jego ograniczoność.

Twierdzenie C.9 Niech A będzie odwzorowaniem liniowym X w Y . Następujące warunki są równoważne:

- A jest ograniczone,
- A jest ciągłe,
- A jest ciągłe w zerze.

Dowód. $1 \Rightarrow 2$: Niech $x_n \rightarrow x$, wtedy $Ax_n \rightarrow Ax$, bo $\|Ax_n - Ax\|_Y = \|A(x_n - x)\|_Y \leq C \|x_n - x\|_X$, a $\|x_n - x\|_X \rightarrow 0$. Implikacja $2 \Rightarrow 3$ jest oczywista. Pokażemy, że $3 \Rightarrow 1$. Niech odwzorowanie A nie będzie ograniczone. Pokażemy, że wtedy nie jest ono ciągłe w 0. Weźmy pewien ciąg x_n , taki, że $\|x_n\|_X = 1$ i jednocześnie:

$$\|Ax_n\|_Y \geq n.$$

Ciąg taki możemy wybrać skoro A nie jest ograniczone. Wtedy ciąg $x_n/\sqrt{n} \rightarrow 0$ i jednocześnie:

$$\left\| A \frac{x_n}{\sqrt{n}} \right\|_Y = \frac{\|Ax_n\|_Y}{\sqrt{n}} \geq \sqrt{n},$$

Zatem ciąg Ax_n/\sqrt{n} nie zbiega do $A0 = 0$, więc odwzorowanie A nie jest ciągłe w zerze. ■

Przestrzeń odwzorowań liniowych ciągłych z normą określoną przez (C-40) oznaczamy przez $L(X, Y)$.

Twierdzenie C.10 Przestrzeń $L(X, Y)$ jest przestrzenią Banacha.

Dowód. Niech A_n będzie ciągiem Cauchy'ego w metryce danej przez normę (C-40). Z nierówności:

$$\|A_n x - A_m x\| \leq \|A_n - A_m\| \|x\|$$

wynika, że $A_n x$ jest ciągiem Cauchy'ego w Y . Z zupełności przestrzeni Y wynika, że ciąg ten ma granicę. Określmy:

$$Ax \equiv \lim_{n \rightarrow \infty} A_n x.$$

A jest odwzorowaniem liniowym, bo:

$$A(\alpha x + \beta x') = \lim_{n \rightarrow \infty} A_n(\alpha x + \beta x') = \alpha \lim_{n \rightarrow \infty} A_n x + \beta \lim_{n \rightarrow \infty} A_n x'.$$

Ustalmy x i $\varepsilon > 0$. Jeśli tylko liczby n i m są odpowiednio duże, mamy $\|A_n - A_m\| \leq \varepsilon$, co pozwala nam wnioskować, że:

$$\|A_n x - A_m x\| \leq \varepsilon \|x\|.$$

Przechodząc do granicy przy $n \rightarrow \infty$ uzyskujemy:

$$\|Ax - A_m x\| \leq \varepsilon \|x\|.$$

Wynika stąd, że $\|Ax\| \leq (\|A_m\| + \varepsilon) \|x\|$ a więc odwzorowanie A jest ograniczone. Element x wybraliśmy dowolnie więc $\|A - A_m\| = \sup_x \left\{ \frac{\|(A - A_m)x\|}{\|x\|} \right\} \leq \varepsilon$, gdzie ε jest dowolną liczbą dodatnią. Wynika stąd zbieżność $A_m \rightarrow A$. ■

C.2.2 Funkcjonały liniowe i przestrzeń dualna

Ważną klasę odwzorowań liniowych stanowią odwzorowania liniowe ciągłe z $(X, \|\cdot\|)$ w \mathbb{R} , nazywane funkcjonalami.

Przestrzeń funkcjonałów liniowych ciągłych na X nazywamy przestrzenią dualną (sprzężoną) do X i oznaczamy przez X^* . Z twierdzenia C.10 wiemy, że X^* jest przestrzenią Banacha. Ważnym zagadnieniem jest określenie czym jest (lub z czym można utożsamić) przestrzeń X^* dla danej przestrzeni X . Twierdzenia odpowiadające na to pytanie nazywane bywają twierdzeniami o reprezentacji.

Niech np. X będzie przestrzenią \mathbb{R}^n z pewną normą (np. $\|\cdot\|_p$). Niech Λ będzie funkcjonalami na X i $x \in X$. Zapiszmy:

$$x = \sum_{j=1}^m x_j e_j,$$

gdzie e_i są wersorami. Z liniowości Λ mamy:

$$\Lambda x = \sum_{i=1}^n x_i \Lambda e_i.$$

Tak więc Λ można jednoznacznie powiązać z elementem $(\Lambda e_1, \dots, \Lambda e_n) \in \mathbb{R}^n$. Łatwo też zauważyć, że odwzorowanie $\alpha \Lambda^1 + \beta \Lambda^2$ odpowiada wektorowi $(\alpha \Lambda^1 e_1 + \beta \Lambda^2 e_1, \dots, \alpha \Lambda^1 e_n + \beta \Lambda^2 e_n)$, a więc także struktura algebraiczna X^* jest taka sama jak \mathbb{R}^n . Matematycy mówią, że przestrzeń X^* jest *izomorficzna* z \mathbb{R}^n .

Poniższe dwa twierdzenia mówią o przestrzeniach sprzężonych z przestrzeniami l^p dla $1 \leq p < \infty$. Długi i techniczny dowód pierwszego z nich pomijamy.

Twierdzenie C.11 Niech $1 < p < \infty$. Przestrzenią sprzężoną z l^p jest przestrzeń l^q , gdzie q jest wykładnikiem sprzężonym z p ($1/p + 1/q = 1$).

Twierdzenie C.12 Przestrzenią sprzężoną z l^1 jest przestrzeń l^∞ .

Dowód. Niech $x \in l^1$, Λ będzie funkcjonałem liniowym ciągłym na l^1 . Niech $e_i \in l^1$ będzie określony jak poprzednio. Szereg $\sum_{n=1}^N x_n e_n$ zbiega do x i z ciągłości i liniowości Λ mamy:

$$\Lambda x = \Lambda \left(\lim_{N \rightarrow \infty} \sum_{n=1}^N x_n e_n \right) = \lim_{N \rightarrow \infty} \sum_{n=1}^N x_n \Lambda e_n.$$

Pokażemy, że ciąg $\Lambda_n = \Lambda e_n$ należy do l^∞ . Mamy:

$$|\Lambda_n| = |\Lambda e_n| \leq \|\Lambda\| \|e_n\| = \|\Lambda\|,$$

skąd wynika, że ciąg $(\Lambda_n)_{n=1}^\infty$ jest ograniczony, a więc należy do l^∞ .

Niech teraz $(\Lambda_n)_{n=1}^\infty \in l^\infty$ i $x \in l^1$. Szereg:

$$\sum_{n=1}^{\infty} \Lambda_n x_n$$

jest zbieżny bezwzględnie, bo:

$$\sum_{n=1}^{\infty} |\Lambda_n x_n| \leq \sum_{n=1}^{\infty} |x_n| \sup_n \{|\Lambda_n|\} = \|x\|_1 \|(\Lambda_n)_{n=1}^\infty\|_\infty.$$

Zatem ciąg $(\Lambda_n)_{n=1}^\infty \in l^\infty$ zadaje na l^1 liniowy funkcjonał postaci:

$$\Lambda x = \sum_{n=1}^{\infty} \Lambda_n x_n. \quad \blacksquare$$

Wydawałoby się, że l^1 jest przestrzenią sprzężoną z l^∞ , jednak tak nie jest. Oczywiście elementy l^1 zadają funkcjonały liniowe na l^∞ , jednak nie wszystkie elementy $(l^\infty)^*$ należą do l^1 .³ Innymi słowy nie wszystkie funkcjonały liniowe na l^∞ mają reprezentację w postaci ($a \in l^\infty$):

$$\Lambda a = \sum_{i=1}^{\infty} \Lambda_i a_i. \quad (\text{C-41})$$

W rozdziale 4 funkcjonały liniowe na przestrzeni dóbr interpretujemy jako system cen. Jeśli funkcjonał ma reprezentację (C-41), wtedy Λ_i naturalnie można interpretować jako cenę i -tego dobra. Jeśli przestrzeń dóbr utożsamiamy z l^∞ system cenowy (np. ten, o którym mowa w tezie drugiego twierdzenia ekonomii dobrobytu) może nie być interpretowany jako „wektor cen”. Intuicja podpowiada, że takie funkcjonały liniowe dużą rolę przykładają dobrom o indeksach zbiegających do nieskończoności. Na szczęście, można w odpowiedni sposób narzucając pewne ograniczenia na preferencje i technologię⁴ zapewnić by system cenowy w równowadze konkurencyjnej miał naturalną dla ekonomisty reprezentację (C-41).

³ Przykład taki można skonstruować w oparciu o twierdzenie Hahna-Banacha (patrz strona 163). Wystarczy skonstruować funkcjonał liniowy nienależący do l^1 na pewnej podprzestrzeni l^∞ i rozszerzyć go na całą l^∞ .

⁴ Patrz Stokey, Lucas i Prescott [13], rozdział 15.3.

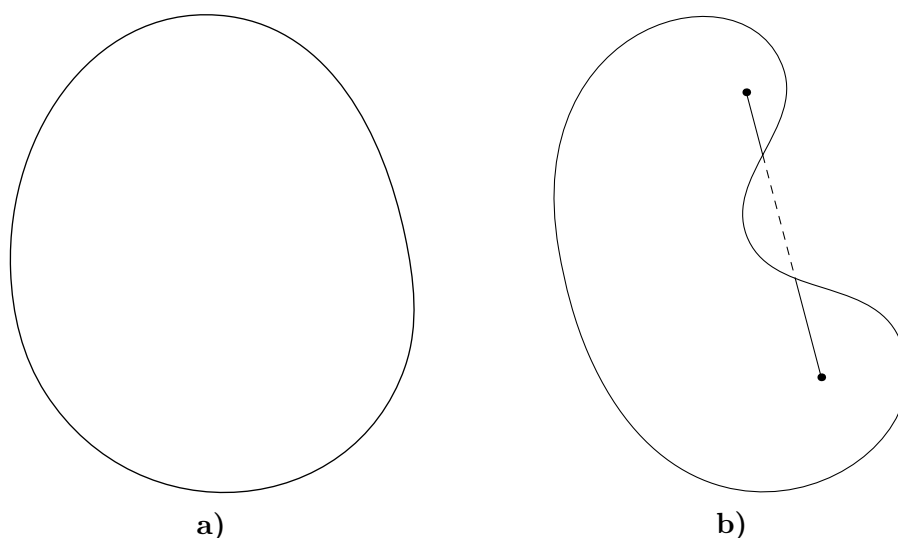
C.3 Wypukłość

Pojęcie wypukłości jest bardzo ważnym pojęciem w matematyce i pojawia się w wielu miejscach w ekonomii jako konsekwencja standardowych założeń o technologii i preferencjach.

Definicja C.18 Niech X będzie przestrzenią liniową. Zbiór $M \subset X$ nazywamy wypukłym, jeśli

$$\forall x, y \in M \forall 0 < \lambda < 1 \quad \lambda x + (1 - \lambda)y \in M. \quad (\text{C-42})$$

Innymi słowy zbiór jest wypukły, jeśli dla dowolnych dwóch jego punktów odcinek je łączący należy do tego zbioru. Przykład zbioru wypukłego i zbioru, który wypukły nie jest zawiera rysunek C.1.



Rysunek C.1: Przykład a) zbioru wypukłego i b) zbioru, który nie jest wypukły

Cechą zbiorów wypukłych, która będzie nas teraz interesować, jest tzw. własność oddzielania. Narysujmy na płaszczyźnie dwa rozłączne wypukłe i domknięte zbiory A i B tak, jak na rysunku C.2a. Zbiory te zawsze można oddzielić linią prostą. W przypadku, gdy choć jeden ze zbiorów wypukły nie jest, to oddzielenie może nie być możliwe, co obrazuje rysunek C.2b.

Linię w \mathbb{R}^2 można zapisać równaniem:

$$\Lambda_1 x_1 + \Lambda_2 x_2 = \gamma,$$

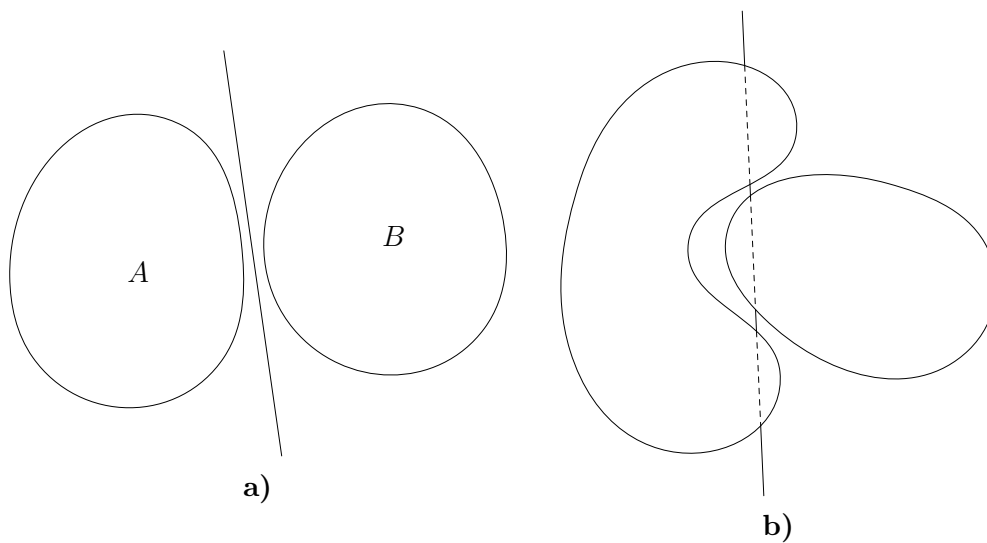
lub następująco:

$$\Lambda x = \gamma,$$

gdzie Λ jest funkcjonałem liniowym na \mathbb{R}^2 , który można utożsamić z wektorem (Λ_1, Λ_2) . Zauważmy, że para Λ i γ dzieli \mathbb{R}^2 na dwie części, do których należą zbiory A i B i zachodzą nierówności:⁵

$$\forall x \in A \quad \Lambda x < \gamma, \quad (\text{C-43})$$

⁵ Nierówności mogą też zachodzić w drugą stronę, ale wtedy można zastąpić Λ przez $-\Lambda$ i γ przez $-\gamma$, co nie zmieni samej prostej, lecz zamieni znaki wyrażenia $\Lambda x - \gamma$ po obu jej stronach.



Rysunek C.2: a) Dwa zbiory wypukłe zawsze można oddzielić funkcjonalem liniowym. b) Jeśli choć jeden ze zbiorów nie jest wypukły, takie oddzielenie może nie być możliwe.

$$\forall_{x \in B} \Lambda x > \gamma. \quad (\text{C-44})$$

Do tej pory rozpatrywaliśmy przypadki dwóch zbiorów wypukłych, domkniętych i rozłącznych w \mathbb{R}^2 .⁶ W przypadku, gdy jeden ze zbiorów nie jest domknięty (dalej oba są wypukłe) oddzielenie w sposób opisany (ostrymi) nierównościami (C-43) i (C-44) może nie być możliwe. Wyobraźmy sobie np. koło otwarte o środku $(-1, 0)$ (zbiór A) i koło domknięte o środku $(1, 0)$ (zbiór B) oba o promieniu 1. Oczywiście zachodzi $A \cap B = \emptyset$. Prosta $(0, y)$ opisana równaniem $\Lambda x = 0$, gdzie $\Lambda_1 = 1$, $\Lambda_2 = 0$ oddziela zbiory A i B , lecz tylko w sensie poniższych nierówności:

$$\forall_{x \in A} \Lambda x < 0,$$

$$\forall_{x \in B} \Lambda x \geq 0.$$

Nasze rozumowanie miało do tej pory charakter wyłącznie heurystyczny. Kilka różnych twierdzeń o oddzielaniu w przypadku przestrzeni \mathbb{R}^n z dowodami można znaleźć np. u Takayamy [22], str. 39-48. Poniżej podajemy najważniejsze z nich.

Twierdzenie C.13 (O oddzielaniu w \mathbb{R}^n) Niech A i B będą niepustymi rozłącznymi zbiorami wypukłymi w \mathbb{R}^n . Wtedy istnieje takie $\Lambda = (\Lambda_1, \dots, \Lambda_n)$ i γ , że $\sum_{i=1}^n \Lambda_i x_i \geq \gamma: x \in A$ i $\sum_{i=1}^n \Lambda_i x_i \leq \gamma: x \in B$.

Twierdzenia o oddzielaniu mają swoje uogólnienia na przypadek nieskończenie-wymiarowy. Uogólnienia te bazują na fundamentalnym twierdzeniu analizy funkcjonalnej – twierdzeniu Hahna-Banacha, które podajemy bez dowodu (patrz Rudin [20], str. 70-71).

⁶ Czytelnik pamiętający z kursu analizy pojęcie zwartości zauważy, że zbiory domknięte i ograniczone w skończeniowym wymiarze są zwarte, zwarty jest także ich iloczyn kartezjański. Odległość $d(x, y)$ dla $x \in A$ i $y \in B$ na mocy twierdzenia Weierstrassa osiąga swoje minimum, które nie może być równe 0 (bo wtedy istniałby punkt, należący i do A i do B). Nieformalnie można powiedzieć, między takimi zbiorami „jest wolna przestrzeń”.

Dodatek C. Podstawy analizy funkcjonalnej

Twierdzenie C.14 (Hahna-Banacha) Niech M będzie podprzestrzenią przestrzeni liniowej X , niech $p: X \rightarrow \mathbb{R}$ spełnia warunki:

$$\forall x, y \in X \quad p(x + y) \leq p(x) + p(y), \quad (\text{C-45})$$

$$\forall x \in X \forall t \geq 0 \quad p(tx) = tp(x). \quad (\text{C-46})$$

Jeśli f jest funkcjonałem liniowym na M spełniającym warunek $fx \leq p(x)$, to daje się rozszerzyć do funkcjonału liniowego Λ na X spełniającego warunki:

$$\forall x \in M \quad \Lambda x = fx, \quad (\text{C-47})$$

$$\forall x \in X \quad -p(-x) \leq \Lambda x \leq p(x). \quad (\text{C-48})$$

Wprowadzimy teraz kilka oznaczeń. Niech X będzie przestrzenią liniową nad \mathbb{R} . Niech $A, B \subset \mathbb{R}$ i $\alpha \in \mathbb{R}$. Wprowadzamy następujące zapisy:

$$A + B \equiv \{a + b: a \in A, b \in B\}, \quad (\text{C-49})$$

i

$$\alpha A \equiv \{\alpha a: a \in A\}. \quad (\text{C-50})$$

Ćwiczenie C.14 Niech A będzie kwadratem o wierzchołkach $(-1, 1)$, $(1, 1)$, $(1, -1)$, $(-1, -1)$, zaś B kołem o promieniu $\frac{1}{2}$ i środku w punkcie $(0, \frac{1}{2})$. Narysuj $A + B$.

Definicja C.19 Zbiór $A \subset X$ nazywamy pochłaniającym jeśli dla każdego $x \in X$ istnieje taka liczba $\alpha > 0$, że $x \in \alpha A$.

Ćwiczenie C.15 Pokaż, że każdy zbiór pochłaniający zawiera wektor zerowy i że każdy otwarty zbiór zawierający zero jest pochłaniający.

Definicja C.20 Niech A będzie zbiorem wypukłym i pochłaniającym. Funkcjonał Minkowskiego zbioru A określamy następująco:

$$\mu_A(x) = \inf\{\alpha > 0: x \in \alpha A\}. \quad (\text{C-51})$$

Ćwiczenie C.16 Niech $X = \mathbb{R}^2$ i A będzie trójkątem o wierzchołkach $(-1, -1)$, $(0, 1)$, $(1, -1)$ Oblicz $\mu_A((0, \frac{1}{2}))$ i $\mu_A((1, 0))$.

Lemat C.2 Niech A będzie zbiorem wypukłym i pochłaniającym. Wtedy:

$$\forall \alpha > 0 \forall x \in X \quad \mu_A(\alpha x) = \alpha \mu_A(x), \quad (\text{C-52})$$

$$\forall x, y \in X \quad \mu_A(x + y) \leq \mu_A(x) + \mu_A(y). \quad (\text{C-53})$$

Dowód. Niech $\alpha > 0$. Z definicji μ_A mamy:

$$\mu_A(\alpha x) = \inf\{\beta > 0: \alpha x \in \beta A\} = \alpha \inf\{\beta/\alpha > 0: x \in \beta/\alpha A\} = \alpha \mu_A(x).$$

Ustalmy liczbę $\varepsilon > 0$. Niech $\alpha = \mu_A(x) + \varepsilon$ i $\beta = \mu_A(y) + \varepsilon$. Wiemy, że $\alpha^{-1}x \in A$ i $\beta^{-1}y \in A$. Z wypukłości zbioru A mamy:

$$z = \frac{\alpha}{\alpha + \beta} \alpha^{-1}x + \frac{\beta}{\alpha + \beta} \beta^{-1}y = (\alpha + \beta)^{-1}(x + y) \in A.$$

Ponieważ $z \in A$, $\mu_A(z) \leq 1$. Stąd i z wcześniej udowodnionego warunku (C-52) wynika, że:

$$\frac{\mu_A(x+y)}{\alpha+\beta} \leq 1,$$

skąd mamy:

$$\mu_A(x+y) \leq \alpha + \beta = \mu_A(x) + \mu_A(y) + 2\varepsilon.$$

Liczbę ε wybraliśmy dowolnie, co daje (C-53). ■

Możemy teraz przystąpić do dowodu twierdzenia o oddzielaniu (geometrycznej wersji twierdzenia Hahna-Banacha)⁷ dla dowolnych przestrzeni.

Twierdzenie C.15 (O oddzielaniu) *Niech X będzie przestrzenią Banacha⁸ a $\in X$ i $B \subset X$, zbiór B będzie wypukły przy czym $a \notin B$ i B ma niepuste wnętrze lub X będzie skończeniowymiarowa. Wtedy istnieje taki funkcjonal liniowy $\Lambda \in X^*$, że dla każdego $b \in B$:*

$$\Lambda a \leq \Lambda b. \quad (\text{C-54})$$

Dowód. W przypadku, gdy X jest skończeniowymiarowa nasza teza jest konsekwencją twierdzenia C.13. Ograniczymy się więc do przypadku, gdy B ma niepuste wnętrze.

Weźmy $b' \in \text{int } B$. Niech $C = b' - B$. Wtedy $0 \in \text{int } C \subset C$. Ponieważ $\text{int } C$ jest pochłaniające (patrz ćwiczenie C.15) pochłaniające jest też C . Funkcjonał Minkowskiego zbioru C spełnia na mocy lematu C.2 warunki (C-45) i (C-46). Zauważmy, że $x' = b' - a \notin C$, skąd wynika, że $\mu_C(x) \geq 1$. Na jednowymiarowej podprzestrzeni M rozpiętej na x' określmy funkcjonal f następująco:

$$fx = \{\alpha: \alpha x' = x\}.$$

Łatwo zauważyć, że f jest funkcjonalem liniowym. Weźmy $\alpha > 0$. Wtedy $\mu_C(x) \geq 1$ i z własności (C-52) wynika, że:

$$f(\alpha x) = \alpha \leq \alpha \mu_C(x) = \mu_C(\alpha x),$$

Oczywiście

$$f(-\alpha x) = -f(\alpha x) < 0 \leq \mu_C(\alpha x).$$

Zatem funkcjonal f spełnia warunki twierdzenia Hahna-Banacha i można go rozszerzyć do takiego funkcjonala Λ na X , że $\Lambda x \leq \mu_C(x)$ i $\Lambda x = f(x): x \in M$. Funkcjonał Λ jest ograniczony, bo w otoczeniu zera $O = \text{int } C \cap -\text{int } C$ jest ograniczony ($x \in O \Rightarrow \mu_C(x) \leq 1$). Weźmy $b \in B$. Mamy:

$$\Lambda(a - b + x') \leq \mu_C(a - b + x') \leq 1,$$

bo $a - b + x' = b' - b \in C$. Z faktu, iż $\Lambda x' = 1$ wynika, że:

$$\Lambda a \leq \Lambda b. \quad \blacksquare$$

⁷ Nie jest to jedyna wersja twierdzenia o oddzielaniu, patrz np. Rudin [20], rozdział 3. Dowody wszystkich twierdzeń o oddzielaniu bazujących na twierdzeniu Hahna-Banacha są jednak bardzo podobne do zamieszczonego poniżej.

⁸ Ogólniej: przestrzenią liniowo topologiczną.

Bibliografia

Indeks

Bibliografia

- [1] P. Aghion, P. Howitt. *Endogenous Growth Theory*. MIT Press, Cambridge, 1998.
- [2] K. J. Arrow. An extension of basic theorems of classical welfare economics. J. Neyman, redaktor, *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, strony 507–532. University of California Press, Berkeley, 1951.
- [3] K. J. Arrow. *General Equilibrium*, wolumen 2 serii *Collected Papers of Kenneth J. Arrow*. Harvard University Press, Cambridge, 1983.
- [4] K. J. Arrow, G. Debreu. Existence of equilibrium for a competitive economy. *Econometrica*, 22:265–290, 1954.
- [5] R. J. Barro, X. Sala-i Martin. *Economic Growth*. MIT Press, Cambridge, 1998.
- [6] R. E. Bellman. *Dynamic Programming*. Princeton University Press, New Jersey, 1957.
- [7] D. Cass. Optimum growth in an aggregative model of capital accumulation. *Review of Economic Studies*, strony 233–240, 1965.
- [8] A. C. Chiang. *Elementy dynamicznej optymalizacji*. Elipsa, Warszawa, 2002.
- [9] G. Debreu. Valuation equilibrium and Pareto optimum. *Proceedings of the National Academy of Sciences*, 40:588–592, 1954.
- [10] G. Debreu. *Mathematical Economics: Twenty Papers of Gerard Debreu*. Cambridge University Press, Cambridge, 1983.
- [11] K. L. Judd. *Numerical Methods in Economics*. MIT Press, Cambridge, 1998.
- [12] T. Koopmans. On the concept of optimal economic growth. *The Economic Approach to Development Planning*. North-Holland, Amsterdam, 1965.

Bibliografia

-
- [13] R. E. Lucas, N. Stokey, E. C. Prescott. *Recursive Methods in Economic Dynamics*. Harvard University Press, Cambridge, 1989.
- [14] A. Mas-Colell, M. D. Whinston, J. R. Green. *Microeconomic Theory*. Oxford University Press, New York, 1995.
- [15] K. Maurin. *Analiza. Część I. Elementy*. Państwowe Wydawnictwo Naukowe, 1971.
- [16] M. J. Miranda, P. L. Fackler. *Applied Computational Economics and Finance*. MIT Press, Cambridge, 2002.
- [17] A. Ralston. *Wstęp do analizy numerycznej*. PWN, Warszawa, 1971.
- [18] F. P. Ramsey. A mathematical theory of saving. *Economic Journal*, strony 543–559, 1928.
- [19] W. Rudin. *Analiza rzeczywista i zespolona*. Wydawnictwo Naukowe PWN, Warszawa, 1998.
- [20] W. Rudin. *Analiza funkcjonalna*. Wydawnictwo Naukowe PWN, Warszawa, 2001.
- [21] T. J. Sargent, L. Ljungqvist. *Recursive Macroeconomic Theory*. MIT Press, Cambridge, 2000.
- [22] A. Takayama. *Mathematical Economics*. Cambridge University Press, Cambridge, wydanie 2, 1985.
- [23] H. R. Varian. *Mikroekonomia*. Wydawnictwo Naukowe PWN, Warszawa, 1995.

Indeks

- algorytm
 - kolejnych przybliżeń, 34
 - rekursywny, 15
- alokacja, 52
 - dopuszczalna, 53
- $\arg \max_x \{f(x)\}$, 7
- $\arg \min_x \{f(x)\}$, 8
- centralny planista, 70
- efektywność w sensie Pareto, 53, 64
- funkcja
 - Lagrange'a, 23, 29
 - produkcji, 73
 - Cobba-Douglasa, 74
 - użyteczności chwilowej, 24
 - CRRA, 81
 - wartości, 12
 - w problemie z dyskontowaniem, 14
- funkcjonał liniowy, 159
- licytator walrasowski, 54
- macierz
 - Hilberta, 130
 - Vandermonde'a, 131
- $\max_x \{f(x)\}$, 7
- metoda
 - kolejnych przybliżeń, 32, 43, 152
 - kolokacji, 91
 - strzelania (ang. *shooting*), 25
- metody
 - projekcyjne, 91
- metryka, 146
- $\min_x \{f(x)\}$, 7
- mnożnik Lagrange'a, 23, 24
- model
 - Ramseya, 30, 42
- nierówność
 - Höldera, 153
 - Minkowskiego, 154
- norma, 145
 - odwzorowania liniowego, 158
- polityka (ang. *policy*), 11
- postęp techniczny, 81
- prawo
 - Walrasa, 55
- przestrzeń
 - Banacha, 148
 - dualna (sprzężona), 159
 - liniowa, 144
 - l^∞ , 146, 149
 - metryczna, 146
 - unormowana, 146
- równanie
 - Bellmana, 12
 - w problemie z dyskontowaniem, 15
 - Eulera, 25, 26, 46
 - polityki, 12
 - w problemie z dyskontowaniem, 15
- równowaga

Indeks

- Arrowa-Debreu, 58
- konkurencyjna, 56
- relacja
 - mocnej preferencji, 53
 - słabej preferencji, 53
- reprezentatywny podmiot, 71
- rozkład
 - LU, 128
- schemat Ponzi'ego, 59
- skrypt MATLAB/OCTAVE, 110
- skrzynka Edgewortha, 52
- stan, 11
- sterowanie (ang. *control*), 11
- supremum, 38
- twierdzenie
 - Banacha o odwzorowaniu zbliżającym, 151
 - Blackwella, 152
 - ekonomii dobrobytu
 - drugie, 67
 - pierwsze, 66
 - Hahna-Banacha, 163
 - o obwiedni, 22
 - o oddzielaniu, 162, 164
- uwarunkowanie macierzy, 129
- warunek
 - NPG, 60
 - transwersalności, 32, 46
- wypukłość
 - preferencji, 64
- zbiór
 - wypukły, 161
- zupełność rynku, 57