

Praca powinna być cytowana jako:

Starosta, B., 2009. Metazbiory. Implementacja relacji i operacji algebraicznych. Rozprawa doktorska.
Polsko-Japońska Wyższa Szkoła Technik Komputerowych.



POLSKO-JAPOŃSKA WYŻSZA SZKOŁA
TECHNIK KOMPUTEROWYCH

Metazbiory

Implementacja relacji i operacji algebraicznych

Bartłomiej Starosta

Praca doktorska wykonana na Wydziale Informatyki
pod kierunkiem prof. dra hab. Witolda Kosińskiego

Warszawa, 2009

Streszczenie W pracy wyłożono podstawy teorii metazbiorów oraz opisano komputerową implementację biblioteki operacji na metazbiorach wraz z przykładowymi aplikacjami demonstrującymi jej zastosowanie.

Teoria metazbiorów definiuje wielowartościową relację należenia elementu do zbioru. W przeciwieństwie do teorii zbiorów rozmytych, jest ona dobrze osadzona w klasycznej teorii mnogości i jest oparta na jej elementarnych pojęciach, posługuje się podobnym językiem. Z drugiej strony, definicje podstawowych relacji i operacji na metazbiorach można wysłowić w sposób umożliwiający ich efektywną implementację w językach programowania.

Oprócz definicji relacji mnogościowych oraz operacji algebraicznych dla metazbiorów, przedstawiono ich podstawowe i najważniejsze właściwości. W szczególności wykazano, że ciało metazbiorów jest algebrą Boole'a. Pokazano też w jaki sposób metazbiory uogólniają pewne zbiory rozmyte.

Teoria metazbiorów łączy dwa odległe światy: abstrakcyjnej teorii matematycznej (silnie bazując na klasycznej teorii mnogości i jej najnowszych wynikach), oraz świat zastosowań komputerowych (poprzez definicje podstawowych pojęć nastawione na efektywność implementacji).

Abstract The dissertation introduces the basis of the theory of metasets. It also includes the computer implementation of metasets operations and sample programs demonstrating its application.

The theory of metasets defines a many-valued membership relation of an element to a set. As oppose to the fuzzy set theory, it is well fitted within the classical set theory and is based on its fundamental notions; it uses a similar language. On the other hand, the definitions of basic relations and operations for metasets may be expressed in the form which enables efficient implementations in programming languages.

Besides the definitions of basic set-theoretic relations and algebraic operations for metasets, their most important properties are investigated. Particularly, it is proved that the field of metasets is a Boolean algebra. Also, a method for representing some type of fuzzy sets by means of metasets is presented.

The metaset theory establishes a bridge between two distant worlds: the world of a pure abstract mathematical theory (being founded on the classical set theory and making use of its contemporary achievements) and the world of machines and computer applications (by means of definitions directed towards the efficiency of implementations).

Spis treści

Cel i zakres pracy	v
1 Wstęp	1
2 Motywacje	5
3 Metazbiory	9
3.1 Oznaczenia	9
3.2 Podstawowe definicje	12
3.2.1 Definicja metazbioru	12
3.2.2 Ważne klasy metazbiorów	14
3.3 Interpretacje metazbiorów	15
3.3.1 Definicja interpretacji	15
3.3.2 Podstawowe fakty	16
4 Działania na metazbiorach	19
4.1 Wymuszanie	19
4.1.1 Definicja wymuszania	20
4.1.2 Przykłady	21
4.2 Podstawowe relacje	22
4.2.1 Należenie warunkowe	22
4.2.2 Równość	24
4.2.3 Zawieranie	26
5 Metazbiory pierwszego poziomu	29
5.1 Podstawowe definicje	30
5.1.1 Poziomy metazbiorów	30
5.1.2 Pokrywanie podzbiorów \mathbb{T}	30
5.1.3 Równoważność podzbiorów \mathbb{T}	32
5.2 Podstawowe relacje	34
5.2.1 Należenie	34
5.2.2 Zawieranie	35
5.2.3 Równość	37
5.3 Operacje algebraiczne	38
5.3.1 Suma	38
5.3.2 Część wspólna	39
5.3.3 Różnica	41
5.4 Właściwości relacji i operacji	47
5.5 Algebra Boole'a	48

5.5.1	Właściwości splotu	48
5.5.2	Ciało metazbiorów	50
5.5.3	Twierdzenie podstawowe	53
6	Metazbiory a zbiory rozmyte	57
6.1	Reprezentacje w \mathfrak{M}^1	57
6.2	Reprezentacje w \mathfrak{M}^1	60
7	Implementacja	61
7.1	Opis biblioteki	63
7.2	Dokumentacja	64
7.3	Dodatkowe aplikacje	64
7.3.1	Pulpit metazbiorów	65
7.3.2	Rozpoznawanie znaków	69
8	Wnioski i zadania na przyszłość	75
	Zawartość płyty	77
	Indeks symboli	79
	Indeks pojęć	81
	Bibliografia	83

Cel i zakres pracy

Praca składa się z dwóch części: teoretycznej i praktycznej. Do tekstu — prezentującego przede wszystkim wyniki teoretyczne — została załączona płytka zawierająca implementację i kod źródłowy biblioteki, dokumentację kodu źródłowego oraz przykładowe aplikacje.

Głównym celem pracy jest zaprezentowanie nowej teorii zbiorów z wielowartościową relacją należenia, oraz wykazanie jej ukierunkowania na zastosowania komputerowe. W związku z tym, oprócz wyników teoretycznych, została zaprezentowana eksperymentalna implementacja i wykorzystujące ją aplikacje.

Część teoretyczna

Teoria metazbiorów definiuje wielowartościową relację należenia elementu do zbioru. Jest dobrze osadzona w klasycznej teorii mnogości i posługuje się podobnym językiem. W szczególności „elementami” metazbiorów są metazbiory. W pracy zdefiniowano podstawowe relacje mnogościowe, jak należenie, równość i zawieranie, oraz operacje algebraiczne: sumę, część wspólną i różnicę. Relacje są reprezentowane przez nieskończoną, przeliczalną liczbę symboli relacyjnych, z których każdy określa jakiś stopień spełnienia relacji, pośredni między prawdą a fałszem. Wprowadzone relacje i operacje mają właściwości zbliżone do ich standardowych odpowiedników z klasycznej teorii mnogości mimo, iż uwzględniają fakt częściowej przynależności elementu do zbioru. W szczególności udowodniono, że operacje algebraiczne spełniają aksjomaty algebry Boole’a.

Oprócz ogólnych definicji relacji, dla pewnej klasy metazbiorów określono ich sformułowania dobrze nadające się do komputerowych implementacji oraz zastosowań. Definicje operacji algebraicznych również są skonstruowane w sposób umożliwiający ich efektywną implementację. Osiągnięto ten cel sprowadzając wykonanie operacji czy testowanie relacji do analizy pewnych związków pomiędzy elementami i podziorami drzewa binarnego, albo w innym ujęciu — do operacji na ciągach binarnych, czyli na ciągach bitów. Należy jednak podkreślić, że tematem niniejszej pracy nie są algorytmy realizujące wymienione operacje, ani też ich złożoność.

Metazbiory uogólniają zbiory rozmyte. Wskazano w jaki sposób można reprezentować pewne szczególnej postaci zbiory rozmyte przy pomocy metazbiorów. Ta szczególna postać nie ogranicza jednak w niczym zastosowań komputerowych opisanej reprezentacji, na których koncentruje się przedstawiona praca.

Część praktyczna

Teoria metazbiorów — w definicjach podstawowych relacji i operacji — jest nastawiona na zastosowania komputerowe. W związku z tym wykonana została eksperymentalna implementacja biblioteki klas reprezentujących metazbiory, umożliwiająca wykonywanie operacji algebraicznych i testowanie relacji. Należy podkreślić, że mimo zakładanego ukierunkowania na efektywność, tematem pracy nie jest wykazanie, że wykonana implementacja jest wydajniejsza niż podobne biblioteki, operujące np. na zbiorach rozmytych. Ma ona przede wszystkim charakter doświadczalny. Możliwość efektywnych implementacji metazbiorów jest zagwarantowana poprzez same konstrukcje definicji relacji i operacji, które można sprowadzić do szybkich operacji na bitach.

Na bazie stworzonej biblioteki klas powstały dwie przykładowe aplikacje. Pierwsza z nich ma charakter edukacyjno-demonstracyjny i służy do wizualizacji struktury metazbiorów oraz reguł rządzących wykonywaniem operacji algebraicznych i spełnianiem relacji.

Druga aplikacja jest próbą zastosowania teorii do rozwiązania praktycznego problemu rozpoznawania pisma (znaków). Program umożliwia definiowanie wieloelementowych wzorców znaków wraz z ocenami jakości poszczególnych elementów wzorca. Jako wynik działania otrzymuje się stopień podobieństwa dostarczonej próbki testowej do zdefiniowanego wcześniej złożonego wzorca. Ten stopień podobieństwa jest reprezentowany przez warunkowe relacje należenia i równości pomiędzy metazbiorami, które reprezentują złożony wzorec i próbkę testową. Program prawidłowo rozpoznaje znaki podobne do zdefiniowanego wzorca i ma tendencję do uogólniania wiedzy zakodowanej w postaci ocen jakości elementów wzorca.

Dowodzi to, że koncepcja metazbiorów sprawdza się w praktycznych zastosowaniach i ma szansę stać się poważną alternatywą dla teorii zbiorów rozmytych konkurując z nią w komputerowym przetwarzaniu nieostrych danych.

Rozdział 1

Wstęp

W życiu codziennym zwykle posługujemy się bardzo nieprecyzyjnymi terminami, jak *dużo*, *mało*, *średnio*, itp. Dla każdego z nas są one zrozumiałe, choć każdy może postrzegać je trochę inaczej. Ich znaczenie silnie zależy również od kontekstu, w którym są użyte. Mimo płynności większości codziennych pojęć, człowiek świetnie radzi sobie z pozyskiwaniem i przetwarzaniem informacji wyrażonej przy ich użyciu.

Z drugiej strony, nauki przyrodnicze i techniczne, a zwłaszcza matematyka, posługują się skrajnie odmiennym językiem, w którym nie ma miejsca na dowolność czy niejednoznaczność. Naukowe wielkości i pojęcia są absolutne i do końca określone, zwykle przy pomocy konkretnych wartości liczbowych. Zawsze i wszędzie oznaczają to samo i tak samo.

Te dwa podejścia wydają się krańcowo przeciwstawne, niemożliwe do pogodzenia. I pewnie pozostałyby w opozycji, gdyby nie potrzeba, która jest matką wynalazków.

Od wielu lat obserwujemy rosnące zainteresowanie formalnymi metodami reprezentacji i przetwarzania nieprecyzyjnej informacji. Z jednej strony jest to spowodowane zalewem informacji i danych, które zwykle należy jakoś przetwarzać, a których większość jest z natury nieprecyzyjna. Z drugiej strony, człowiek jako końcowy odbiorca przetworzonych informacji, często oczekiwałby zinterpretowanych danych, w postaci „rozmytych” kategorii, a nie suchych liczb. Wyobraźmy sobie, że pogodnyka w prognozie zamiast informacji typu „jutro będzie ciepło, ale weźcie z domu parasole” podaje nam tajemniczy, niewiele mówiący ciąg liczb, przedstawiający rozmaite parametry atmosferyczne, reprezentujące przewidywany pomiar z dnia następnego.

Właściwie, potrzeba formalnego reprezentowania niedokładnych danych jest znacznie starsza i ma o wiele głębsze korzenie. Jest chyba tak stara jak nauka — zwłaszcza te jej gałęzie, które czerpią dane z obserwacji i pomiarów — i jest w nią permanentnie wpisana. Świat, w którym żyjemy jest całkowicie nieprecyzyjny, nie posiadający sztywnych, jasno określonych granic. Dotyczy to praktycznie każdego jego aspektu, za wyjątkiem matematyki, która z definicji ma być krańcowo precyzyjna. Rozmaite gałęzie nauki, szczególnie społeczne, przyrodnicze czy techniczne próbują uczynić go w sztuczny sposób precyzyjnym poprzez abstrakcję pojęć i przybliżanie rozmaitych wielkości. Robią to, aby umożliwić następnie przetworzenie zgromadzonych nieostrych danych, np. pochodzących z pomiarów, precyzyjnymi metodami matematycznymi.

Czy jest to rzeczywiście konieczne, skoro nawet tak konkretna nauka jak fizyka w rzeczywistości nie może dysponować wielkościami całkowicie precyzyjnymi? Każdy pomiar jest obarczony jakimś błędem, co jest może oczywiste kiedy używa się linijki do zmierzenia długości kijka, który zresztą nigdy nie jest wycinkiem prostej. Ale jest to także naukowym faktem sformułowanym w postaci zasady nieoznaczoności Heisenberga, która wskazuje granice dokładności z jaką możemy poznać (domniemany?) stan faktyczny. Na poziomie makroskopowym żaden pomiar nie jest dokładny w sensie matematycznym, co jest m.in. konsekwencją nieciągłości materii. Czy jesteśmy w stanie zmierzyć, że np. temperatura w jakimś punkcie x jest równa dokładnie y , a nie y z dokładnością do ε ? Z pewnością nie, bo sam fakt pomiaru temperatury jakiegoś obiektu wpływa na jego stan energetyczny. A nawet gdyby dokładne określenie temperatury byłoby możliwe, to przecież za chwilę w rozważanym punkcie ta temperatura się zmieni, więc pomiar należałoby umiejscowić dokładnie w czasie. Ale jak, skoro przecież w momencie dokonania odczytu zmierzona wartość jest już nieaktualna, gdyż świat ulega nieustannym przemianom? Tak więc musimy się zadowolić wielkościami przybliżonymi i nauczyć na nich operować.

Niewątpliwie, zapotrzebowanie na teorie i mechanizmy umożliwiające reprezentację i przetwarzanie niedokładnych danych nie jest tak nowe jak by się mogło wydawać. Nie jest wyłącznie produktem ery komputerów produkujących nieprzebrane morze informacji, ale jest na stałe wpisane w relację między obserwatorem i obserwowanym, pomiędzy mierzącym, mierzonym i wynikiem pomiaru. Statystyka jest w pewnym zakresie dziedziną wychodzącą naprzeciw owym oczekiwaniom, z powodzeniem stosowaną od niemalże stu lat, a przecież nie powstała ona nagle, tylko była wynikiem długotrwałego procesu, który z kolei był wywołany przez określone, jeszcze wcześniejsze potrzeby. Możemy zatem sformułować tezę, że formalne metody matematyczne umożliwiające reprezentację i przetwarzanie nieostrej informacji były i są potrzebne wszędzie w nauce, i zapewne będą w powszechnym użyciu kiedy tylko osiągną odpowiedni stan dojrzałości.

Niemówność dysponowania dokładnymi informacjami wejściowymi — cokolwiek by to nie znaczyło — nie jest jedynym argumentem za teoriami przetwarzania nieprecyzyjnych danych. Warto w tym miejscu przytoczyć zasadę niespójności Zadeha: „w miarę wzrostu złożoności systemu nasza zdolność do formułowania istotnych stwierdzeń dotyczących jego zachowania maleje osiągając w końcu próg, poza którym precyzja i istotność stają się cechami wzajemnie prawie się wykluczającymi” ([35], cytata za [24]). A więc po przekroczeniu pewnej bariery złożoności dokładny system staje się niemożliwy do kontrolowania, nawet komputerowo, a zatem właściwie bezużyteczny.

Wspomnieliśmy na początku, że ludzki umysł ma zadziwiającą zdolność przetwarzania nieostrej informacji. Dlaczego nie potrafią tego maszyny? Czy można je również tego nauczyć? Komputery jako maszyny matematyczne, oparte są na tradycyjnym, kartezjańskim modelu nauki, w którym przetwarzana informacja musi być precyzyjnie określona, zwykle przy pomocy konkretnych wartości liczbowych. Jednak gdybyśmy dysponowali formalną teorią umożliwiającą reprezentowanie nieprecyzyjnych danych i mechanizmem pozwalającym je przetwarzać, to można by nauczyć tego również maszyny, np. za pomocą odpowiedniego programu komputerowego.

Dzisiejszy świat jest zalany ogromną masą informacji, której charakter jest w większości rozmyty. Człowiek nie jest w stanie przyswoić ani przeanalizować

tak dużej ilości, niezbędne jest wspomaganie maszynowe. Stosowane obecnie metody przetwarzania nieostrych danych przez maszyny są przeważnie w dalszym ciągu staroświeckie.¹ Polegają one na wstępnej klasyfikacji (wyostrzeniu) danych, zwykle już na etapie pomiarów, w wyniku których reprezentuje się wielkość z natury nie dającą się dokładnie określić przy pomocy dokładnej wartości (np. temperatura, wysokość zbiorów płodów rolnych, poparcie dla partii politycznych, itp.). Następnie przetwarza się wstępnie przystosowane dane tradycyjnymi, precyzyjnymi metodami uzyskując wynik dokładny na tyle, na ile pozwalają na to komputery, które swoją drogą wprowadzają pewien błąd w obliczeniach zmiennopozycyjnych. Na końcu prezentuje się uzyskany konkretny wynik w formie zinterpretowanej, zrozumiałej dla człowieka. Na przykład w postaci jakiejś rozmytej kategorii: *jest dobrze, plony są takie jak w poprzednim roku, mamy duże poparcie*, itp.

Powstaje pytanie: czy można stworzyć mechanizm potrafiący bezpośrednio operować na pozyskiwanej nieprecyzyjnej informacji tak, by nie było potrzeby wstępnego przystosowywania danych wejściowych do tradycyjnych, dokładnych mechanizmów, który to proces nieuchronnie prowadzi do utraty informacji i wprowadzenia chaosu? Taki mechanizm, pozwoliłby — co może zabrzmieć dziwnie — na dokładniejsze reprezentowanie i przetwarzanie niedokładnych danych przez maszyny, odciążając tym samym człowieka od potrzeby przygotowywania danych wejściowych do postaci zrozumiałej przez maszynę a następnie tłumaczenia wyników będących odpowiedzią w języku maszyny, na język ludzki.

Dochodzimy w ten sposób do dwóch wniosków. Po pierwsze widzimy, że całkowicie precyzyjna i jednoznaczna nauka, zwłaszcza stosowana, to — za wyjątkiem matematyki — czysta utopia. Fakt ten jest na zawsze wpisany w niedokładność danych pozyskiwanych czy to z pomiarów lub obserwacji, czy też w jakikolwiek sposób inny niż wymyślone matematyczne koncepcje. Uzmysłwienie sobie tej oczywistej prawdy czyni potrzebę stworzenia odpowiednich mechanizmów i teorii służących do reprezentacji takich danych jeszcze bardziej palącą. Po drugie, w dobie zalewu ogromem informacji większość danych musi być przetwarzana maszynowo. Powstaje zatem dodatkowy wymóg dla takiej teorii: musi ona dać się efektywnie implementować w dzisiejszych komputerach.

Przedstawiona w niniejszej pracy teoria metazbiorów wychodzi naprzeciw tym potrzebom. Z jednej strony rozszerza klasyczną teorię mnogości o wielowartościową relację należenia. Z drugiej strony robi to formułując definicje przy użyciu dyskretnych (tzn. nie ciągłych) pojęć, akceptowalnych przez maszyny.

¹Oczywiście, nie mamy tu na myśli metod sztucznej inteligencji, np. opartych na logice rozmytej.

Rozdział 2

Motywacje

Klasykzna teoria mnogości Zermelo-Fraenkla (ZFC) ([18]), która wykrystalizowała się w latach dwudziestych XX wieku, stanowi podstawę współczesnej matematyki ([25]), a w konsekwencji i nauk przyrodniczych — wszystkie powszechnie używane teorie matematyczne można do niej sprowadzić. Mimo niezaprzeczalnych zalet, niesie ona ze sobą pewne ograniczenia, które utrudniają, a nierzadko całkowicie uniemożliwiają jej stosowanie do opisu rzeczywistych zjawisk. Te ograniczenia sprowadzają się do ostrego podziału na dwie kategorie: *prawda* bądź *fałsz*. Albo: *należy* — *nie należy*.

Dlatego już na początku ubiegłego wieku powstały pierwsze koncepcje wyjścia poza ten pojęciowy dualizm w postaci prac Łukasiewicza, twórcy pierwszej logiki wielowartościowej ([19], [20]). Rozszerzył on zbiór dopuszczalnych wartości logicznych o trzecią wartość *nie wiem*, a następnie o dowolną skończoną liczbę wartości pośrednich pomiędzy prawdą a fałszem. Uogólnił później ten model na nieskończoną, przeliczalną liczbę wartości logicznych pomiędzy 0 i 1, reprezentowanych przez liczby wymierne. W okresie międzywojennym wielkie zasługi dla rozwoju logik wielowartościowych (jak również teorii mnogości i logiki w ogóle) położyła Polska Szkoła Logiczna ([21]).

Kolejnym ważnym etapem na drodze poszukiwań wyjścia poza binarny schemat myślenia było pojawienie się w latach sześćdziesiątych ubiegłego wieku koncepcji zbiorów rozmytych zaproponowanej przez Zadeha ([34]), która wzbogaca relację należenia elementu do zbioru o kontinuum wartości pośrednich pomiędzy prawdą i fałszem. Zbiór rozmyty A na przestrzeni X jest określony przez swoją funkcję przynależności $\mu_A: X \mapsto I$, która każdemu elementowi $x \in X$ przyporządkowuje wartość liczbową z przedziału jednostkowego, reprezentującą jego stopień należenia do A . W ten sposób zbiór rozmyty określa również stopień nieprzynależności elementu do zbioru, którego wartość jest dopełnieniem do jedności stopnia przynależności.

Wkrótce potem Goguen ([14]) uogólnił funkcję przynależności tak, by mogła przyjmować wartości w dowolnej algebrze lub kracie L uzyskując w ten sposób zbiór L -rozmyty (ang. L -fuzzy set). Zbiory L -rozmyte są pod względem konstrukcji bliższe przedstawionym tu metazbiorom niż standardowe zbiory rozmyte, jednak to podobieństwo nie jest zbyt daleko idące. Poza ogólną koncepcją reprezentowania związków pomiędzy stopniami przynależności w postaci kraty (w przeciwieństwie do liniowego porządku dla zwykłych zbiorów rozmytych), nie uwzględnia żadnych innych właściwości metazbiorów.

Zbiory rozmyte typu 1, których funkcje przynależności przyjmują konkretne wartości (czy to liczbowe, czy też w kracie), tak naprawdę nie modelują niepewności stopnia należenia elementu do zbioru, gdyż ten stopień jest jednak konkretnie określony, choć może różny od pewności. Dlatego w latach siedemdziesiątych Zadeh ([36]) wprowadził zbiory rozmyte typu 2, których funkcje przynależności przyjmują wartości będące same w sobie zbiorami rozmytymi. W szczególności wartość funkcji przynależności może być przedziałem zawartym w odcinku nieostrokowym. Zbiór rozmyty będący wartością funkcji przynależności wyraża nieokreśloność stopnia należenia elementu do zbioru rozmytego.

W latach osiemdziesiątych Atanassow rozszerzył model zbiorów rozmytych dodając postulat mówiący, że poziom nieprzynależności nie musi być dopełnieniem do jedności stopnia przynależności. Powstały w ten sposób intuicjonistyczne zbiory rozmyte ([2], [3]), w których oprócz funkcji przynależności μ , mamy jeszcze osobną funkcję nieprzynależności ν , przy czym ich suma może być mniejsza od jedności. Różnica $1 - (\mu(x) + \nu(x))$ nazywana jest intuicjonistycznym indeksem rozmytym albo poziomem wahania, i wyraża stopień niepewności co do relacji (nie)należenia elementu x do zbioru.

Pojawienie się i rozwój zbiorów rozmytych spowodowało renesans zainteresowań logikami wielowartościowymi ([7], [8]). Już pod koniec lat sześćdziesiątych Zadeh wprowadził logikę rozmytą opartą na zbiorach rozmytych ([37]). Próbuje ona naśladować sposób rozumowania człowieka, który wyciąga wnioski ze słabo określonych przesłanek, w szczególności umożliwia przetwarzanie informacji określonej w języku naturalnym (koncepcja zmiennej lingwistycznej, [38]). Logikę rozmytą z powodzeniem stosuje się w przemysłowych systemach sterowania każdej skali ([24]), jak również w elektronicznych urządzeniach powszechnego użytku. Związki między zbiorami rozmytymi a logiką rozmytą są przedstawione w ([13]).

Warto w tym miejscu wspomnieć również o alternatywnych teoriach zbiorów, opartych na teorii Zermelo-Fraenkla. Teoria półzbiorów (cz. polomnožina, ang. semiset) stworzona przez Vopenkę i Hajka w latach siedemdziesiątych XX wieku ([32]), pozwala reprezentować zbiory z nieostrymi granicami. Związki pomiędzy teorią półzbiorów i zbiorów rozmytych są przedstawione w ([22]). Teoria zbiorów przybliżonych (ang. rough set), zaproponowana przez Pawłaka na początku lat osiemdziesiątych ([23]) przybliża zbiór przez jego ograniczenia: górne i dolne, które w pierwotnym ujęciu są klasycznymi (ostrymi) zbiorami, ale mogą być również zbiorami rozmytymi ([12]).

Teoria zbiorów rozmytych po latach spotyka się z rosnącym zainteresowaniem, które można wytłumaczyć naturalnością wyjściowej koncepcji, a przede wszystkim lawinowo rosnącym zapotrzebowaniem na narzędzia sztucznej inteligencji ze strony nauki, przemysłu i biznesu. Wynika ono ze wspomnianych wcześniej trudności ze stosowaniem tradycyjnego, dwuwartościowego podejścia do opisu zjawisk nieostrych i przetwarzania nieprecyzyjnych danych, oraz jest związane z rosnącą złożonością stosowanych systemów.

Wychodząc naprzeciw tym potrzebom, bazując na swojej znajomości podstaw matematyki, a w szczególności klasycznej teorii zbiorów, jak również na doświadczeniu programistycznym i wiedzy dotyczącej zasad funkcjonowania komputerów ([4], [5], [6], [27]), zaproponowałem wpisującą się w omawiany nurt teorię metazbiorów. Łączy ona elegancję abstrakcyjnej teorii matematycznej, jaką bez wątpienia jest ZFC, z wymaganiami efektywności implementacji narzuconym przez dynamicznie rozwijający się obszar zastosowań, zwłaszcza kompu-

terowych.

Koncepcja metazbioru ma swoje źródła w forsingu ([11]) wynalezionym na początku lat sześćdziesiątych przez kanadyjskiego matematyka P. Cohena. Jest to technika stosowana w teorii mnogości Zermelo-Fraenkla do dowodzenia niezależności aksjomatów. Udało się dzięki niej rozwiązać m.in. problem hipotezy continuum ([9], [10]), również uogólnionej, i wiele innych.¹ W forsingu jednym z podstawowych pojęć jest \mathbb{P} -nazwa (ang. \mathbb{P} -name), gdzie \mathbb{P} oznacza pewien częściowy porządek. Zbiór, który jest \mathbb{P} -nazwą traktuje się jak „nazwę” na zbiór, którego w danym modelu nie ma, ale który może z niej powstać dzięki zastosowaniu odpowiedniego klucza (ultrafiltra). Właściwie, zbiorów, które opisuje \mathbb{P} -nazwa, które mogą z niej powstać, jest wiele, bo można stosować różne klucze do wycinania zbiorów z \mathbb{P} -nazwy, a więc jest ona reprezentantem wielu zbiorów jednocześnie. Mimo, że na poziomie modelu, do którego należy \mathbb{P} -nazwa, nie wiemy jak dokładnie będą wyglądać zbiory z niej wyprodukowane, możemy wywnioskować różne ich właściwości na podstawie budowy porządku \mathbb{P} , który jest znany w danym modelu. Do wnioskowania służy relacja wymuszania zachodząca pomiędzy elementem porządku \mathbb{P} a formułą teorii mnogości.

W tym miejscu analogia między forsingiem i metazbiorami się kończy. Co prawda definicja metazbioru jest właściwie szczególnym przypadkiem definicji \mathbb{P} -nazwy (bo zamiast częściowego porządku \mathbb{P} występuje drzewo binarne \mathbb{T}), a relacja wymuszania została przeniesiona na grunt metazbiorów niemal dosłownie, to jednak zupełnie inny jest obszar zastosowań i pole działania. Mechanizm forsingu działa w świecie zbiorów nieskończonych, w większości nieprzeliczalnych. Teoria metazbiorów mimo, że oczywiście może również opisywać zbiory nieskończone, w warstwie aplikacyjnej koncentruje się na zbiorach skończonych, bo takie dają się reprezentować w komputerach. Celem stosowania forsingu jest rozszerzenie modelu dla fragmentu ZFC, w którym się rzecz odbywa, o jakiś nieobecny w nim zbiór, który nada nową właściwość powstałemu w ten sposób nowemu modelowi (np. doda jakąś funkcję ustalającą równoliczność zbiorów, które w wyjściowym modelu nie były równoliczne). Metazbiór jest sam w sobie zbiorem z wielowartościową relacją należenia, którą udało się zdefiniować dzięki relacji wymuszania, przenosząc sposób należenia elementów zbiorów, które dany metazbiór reprezentuje, na tenże metazbiór.

Oprócz zapożyczonych z forsingu koncepcji postrzegania wielu zbiorów poprzez jeden „zbiór” (\mathbb{P} -nazwa, metazbiór) i możliwości formułowania właściwości tego „zbioru” na podstawie właściwości reprezentowanych przez ten zbiór przy pomocy relacji wymuszania, do teorii metazbiorów przeniknęła jeszcze tylko tu i ówdzie terminologia i oznaczenia.

W czym przejawia się komputerowe ukierunkowanie metazbiorów? Jak zobaczymy w rozdziale 5, podstawowe relacje i operacje można sformułować w terminach związków pomiędzy elementami i podzbiarami drzewa binarnego. Każdy taki element może być reprezentowany jako ciąg zer i jedynek. Albo ciąg bitów. W związku z tym, sprawdzenie czy jakieś dwa elementy drzewa są porównywalne, albo inaczej mówiąc leżą na tej samej gałęzi, może zostać sprowadzone do sprawdzenia, czy jeden ciąg jest prefiksem drugiego, albo po prostu do pewnej liczby przesunięć bitowych. Operacje na bitach są wykonywane bardzo szybko — kompilatory zwykle optymalizują mnożenie i dzielenie przez potęgę dwójki za pomocą właśnie przesunięć bitowych ([1]). To daje nadzieję nie tylko

¹Paul Cohen został za nią uhonorowany Medalem Fieldsa w 1966 r.

na uzyskanie efektywnej implementacji w odpowiednim języku programowania, ale również stanowi furtkę do stworzenia implementacji sprzętowej, opartej na szybkich, wyspecjalizowanych układach elektronicznych, podobnie jak dla sieci neuronowych.

Oczywiście, pozostaje jeszcze kwestia złożoności obliczeniowej zdefiniowanych operacji na metazbiorach. Nie była ona na razie gruntownie badana, jednak pobieżna analiza i doświadczenia uzyskane w trakcie prac nad pierwszą wersją biblioteki² wskazują na akceptowalną złożoność rzędu wielomianowego (najwyżej kwadratową), którą można uzyskać tanim kosztem, tzn. bez potrzeby projektowania wyrafinowanych algorytmów, a po prostu mniej lub bardziej dosłownie zapisując formuły matematyczne z definicji operacji bezpośrednio w języku programowania. Warto na zakończenie zauważyć, że niezależnie od wewnętrznej złożoności obliczeniowej tych operacji nie ma możliwości zmiany ich definicji, gdyż są one zdeterminowane przez relację wymuszania i muszą pozostać takimi, jakie są.

Tak więc implementacje metazbiorów będą wydajne. Co prawda, ideał nie został jeszcze (bo nie miał być) osiągnięty w bieżącej, eksperymentalno-szkolnej wersji, która jest nastawiona przede wszystkim na czytelność kodu źródłowego. Jednak w trakcie prac nad nią potwierdziły się teoretyczne założenia, które znajdują urzeczywistnienie w przyszłej, docelowej implementacji.

Z drugiej strony, wprost z definicji zbioru rozmytego wynika, że operacje na nich muszą w taki czy inny sposób angażować operacje na liczbach zmiennopozycyjnych. Takie działania — z punktu widzenia komputera — są dość kosztowne czasowo w porównaniu z operacjami na liczbach całkowitych czy wręcz na bitach, i z tego powodu negatywnie wpływają na efektywność każdej implementacji operacji na zbiorach rozmytych. A przecież stosowanie liczb zmiennopozycyjnych nie jest konieczne. W komputerach nie możemy reprezentować funkcji ciągłych o wartościach rzeczywistych, bo mamy do dyspozycji wyłącznie zmienne typów przyjmujących skończoną liczbę możliwych wartości. Ograniczając się do liczb dwójkowo-wymiernych, odpowiednio przybliżając ciągłe funkcje przynależności funkcjami kawałkami stałymi o takich wartościach, nie tracimy teoretycznej ogólności, a zwiększymy wydajność komputerowych aplikacji.

To spostrzeżenie stanowi kanwę dla przedstawionej w rozdziale 6 metody reprezentacji zbiorów rozmytych za pomocą metazbiorów. Było ono również pierwotnym bodźcem, który doprowadził do sformułowania operacji na metazbiorach w sposób przedstawiony w rozdziale 5, czyli w języku zrozumiałym dla maszyn. Jednak ogólne definicje relacji, tak jak zostały wprowadzone w rozdziale 4, mają swoje korzenie w forsingu.

²Biblioteka wraz z kodem źródłowym znajduje się na załączonej płycie.

Rozdział 3

Metazbiory

W tym rozdziale podamy definicję metazbioru, wyróżnimy najważniejsze ich rodzaje oraz wprowadzimy technikę interpretacji, która później umożliwi określenie podstawowych relacji.

Najpierw jednak, dla ustalenia uwagi, przypomnimy kilka znanych pojęć i faktów, oraz wprowadzimy używane dalej oznaczenia.

3.1 Oznaczenia

Symbolem $|x|$ będziemy oznaczać moc zwykłego zbioru x , \aleph_0 to moc zbioru liczb naturalnych \mathbb{N} . Liczby naturalne to skończone liczby porządkowe, zatem $0 \in \mathbb{N}$, bo najmniejszą liczbą porządkową jest $0 = \emptyset$. Kolejne skończone liczby porządkowe są postaci $\alpha \cup \{\alpha\}$. I tak: $1 = \{\emptyset\}$, $2 = \{0, 1\} = \{\emptyset, \{\emptyset\}\}$, itd.

Częściowe porządki Relacja \leq jest *częściowym porządkiem* na zbiorze \mathbb{P} , jeśli dla dowolnych $p, q, r \in \mathbb{P}$ spełnia następujące warunki:

$$\begin{aligned} p &\leq p, && \text{(zwrotność)} \\ (p \leq q \wedge q \leq p) &\rightarrow p = q, && \text{(antysymetryczność)} \\ (p \leq q \wedge q \leq r) &\rightarrow p \leq r. && \text{(przechodność)} \end{aligned}$$

Element $p \in \mathbb{P}$ jest *elementem największym* w \mathbb{P} jeśli $\forall_{q \in \mathbb{P}} q \leq p$. Natomiast jeśli $\forall_{q \in \mathbb{P}} (q \geq p \rightarrow q = p)$, to p jest *elementem maksymalnym* w \mathbb{P} . Podobnie definiujemy element *najmniejszy* i *minimalny*.

Częściowy porządek z elementem największym jest to trójka $\langle \mathbb{P}, \leq, \mathbb{1} \rangle$, gdzie \mathbb{P} jest zbiorem, \leq jest relacją częściowego porządku na \mathbb{P} , a $\mathbb{1}$ to element największy w \mathbb{P} .

Drzewo binarne Dla $n \in \mathbb{N}$, oznaczmy przez ${}^n 2 = \{f: n \mapsto 2\}$, przy czym $n = \{0, \dots, n-1\}$ a $2 = \{0, 1\}$. Zatem ${}^n 2$ to zbiór funkcji o wartościach binarnych, których dziedziną jest liczba porządkowa n , a przeciwdziedziną liczba porządkowa 2. Inaczej mówiąc, jest to zbiór ciągów zero-jedynkowych długości n . Widać, że ${}^0 2 = \{\emptyset\}$ zawiera tylko funkcję pustą, oraz ${}^1 2 = \{\langle 0, 0 \rangle, \langle 0, 1 \rangle\}$ zawiera dwa ciągi długości 1, więc $|{}^0 2| = 1$ i $|{}^1 2| = 2$.

Symbolem ${}^{<\mathbb{N}}2$ oznaczamy zbiór wszystkich funkcji o wartościach binarnych, których dziedzinami są skończone liczby porządkowe:

$${}^{<\mathbb{N}}2 = \bigcup_{n \in \mathbb{N}} {}^n 2 . \quad (3.1.1)$$

W zbiorze ${}^{<\mathbb{N}}2$ wprowadzamy relację częściowego porządku w następujący sposób. Jeśli $f, g \in {}^{<\mathbb{N}}2$, $f: n \mapsto 2$ i $g: m \mapsto 2$, to $f \leq g$, gdy $f \supseteq g$, czyli $n \geq m$ i $f \upharpoonright m = g$.¹ Zatem relacja \leq to odwrotne zawieranie się funkcji traktowanych jako zbiory par uporządkowanych.

Drzewo binarne to częściowy porządek z elementem największym $\langle \mathbb{T}, \leq, \mathbb{1} \rangle$, gdzie $\mathbb{T} = {}^{<\mathbb{N}}2$, relacja \leq to odwrotne zawieranie się funkcji, a element największy $\mathbb{1}$, nazywany *korzeniem* drzewa \mathbb{T} , to \emptyset — funkcja pusta. W dalszym ciągu będziemy utożsamiać \mathbb{T} z $\langle \mathbb{T}, \leq, \mathbb{1} \rangle$.

Elementy drzewa \mathbb{T} będziemy oznaczać literami $p, q, r \dots$, i będziemy je nazywać *warunkami*. Jeśli $p, q \in \mathbb{T}$, to mówimy, że warunek p jest *silniejszy* od warunku q , gdy $p \leq q$ (czyli $p \supseteq q$ jako funkcje), oraz że warunek p jest *słabszy* od warunku q , gdy $p \geq q$. Warunek silniejszy, to większa funkcja, niosąca więcej informacji. Funkcja pusta \emptyset , będąca korzeniem drzewa, jest zawarta w każdej funkcji, więc jest warunkiem najslabszym.

Jeśli $p, q \in \mathbb{T}$ są porównywalne, to oznaczamy ten fakt symbolem $p \top q$. Jeśli są one nieporównywalne, to przyjmujemy oznaczenie $p \perp q$. Dla porównywalnych $p, q \in \mathbb{T}$ przyjmujemy:

$$\min(p, q) = \begin{cases} p & \text{gdy } p \leq q , \\ q & \text{gdy } p > q , \end{cases} \quad (3.1.2)$$

oraz

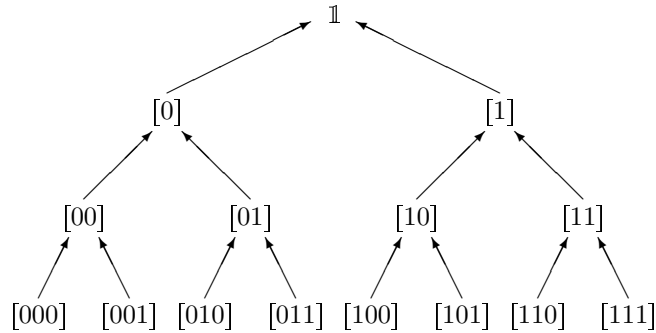
$$\max(p, q) = \begin{cases} p & \text{gdy } p \geq q , \\ q & \text{gdy } p < q . \end{cases} \quad (3.1.3)$$

Zbiór funkcji ${}^{<\mathbb{N}}2$ można postrzegać jako zbiór skończonych ciągów zerojedynkowych, włączając w to ciąg pusty \emptyset . W związku z tym elementy drzewa binarnego \mathbb{T} będziemy oznaczać w następujący sposób (por. rys. 3.1):

- $[i_0 i_1 \dots i_k]$ gdzie $i_j \in \{0, 1\}$ (np. $[011]$ to trójelementowy ciąg binarny zaczynający się od 0),
- $[\]$ jest ciągiem pustym — korzeniem,
- jeśli $p \in \mathbb{T}$ to $p \cdot 0$ i $p \cdot 1$ oznaczają bezpośrednich potomków wierzchołka p , w szczególności:

$$\begin{aligned} [\] \cdot 0 &= [0] , \\ [01] \cdot 0 &= [010] , \\ [1] \cdot 0 \cdot 1 &= [101] . \end{aligned}$$

¹Taki sposób skierowania relacji \leq może wydać się dziwny, jednak stanie się jasny, kiedy okaże się, że element największy porządku $\mathbb{1}$ reprezentuje pełne należenie do metazbiory, podczas gdy elementy mniejsze w tej relacji oznaczają mniejsze stopnie należenia, spowodowane nałożeniem silniejszych, bardziej restrykcyjnych warunków na należenie, które trudniej spełnić.



Rysunek 3.1: Poziomy 0–3 i uporządkowanie warunków w drzewie binarnym \mathbb{T} . Strzałki wskazują element większy, czyli słabszy warunek

Antylańcuchy, gałęzie, poziomy Zbiór $C \subset \mathbb{T}$ jest *łańcuchem* w \mathbb{T} , jeśli składa się z elementów parami porównywalnych:

$$\forall_{p,q \in C} (p \leq q \vee q \leq p) . \quad (3.1.4)$$

Zbiór $A \subset \mathbb{T}$ jest *antyłańcuchem* w \mathbb{T} jeśli składa się z elementów parami nieporównywalnych:

$$\forall_{p,q \in A} (p \neq q \rightarrow p \perp q) . \quad (3.1.5)$$

Gałąź w drzewie \mathbb{T} to łańcuch maksymalny, czyli taki łańcuch, którego nie da się rozszerzyć poprzez dodanie elementów tak, by zachowany był warunek wzajemnej porównywalności.

Antylańcuch maksymalny to taki antylańcuch, którego nie da się rozszerzyć — jest elementem maksymalnym ze względu na relację zawierania się antylańcuchów. Dodając element do antylańcucha maksymalnego sprawiamy, że przestaje on być antylańcuchem.

FAKT 3.1.1. Jeśli antylańcuch przecina wszystkie gałęzie w \mathbb{T} , to jest maksymalny.

Na rys. 3.1 elementy $\{[00], [01], [100]\}$ tworzą przykładowy antylańcuch, natomiast $\{[0], [1]\}$ i $\{[00], [01], [1]\}$, a nawet samo $\{1\}$ są antylańcuchami maksymalnymi.

Niech $p \in \mathbb{T}$. Jeśli C jest antylańcuchem takim, że $\forall_{q \in C} (q \leq p)$, to powiemy, że C jest antylańcuchem *poniżej* p . Jeśli C jest maksymalny wśród antylańcuchów o tej własności, tzn. leżących poniżej p , to powiemy, że C jest *antyłańcuchem maksymalnym poniżej* p . W takiej sytuacji dodanie jakiegokolwiek nowego elementu $q \leq p$ do C spowoduje, że przestanie on być antylańcuchem.

FAKT 3.1.2. Jeśli antylańcuch poniżej p przecina wszystkie gałęzie zawierające p , to jest maksymalny poniżej p .

Dla ustalonego $n \in \mathbb{N}$, zbiór wszystkich funkcji $n \mapsto 2$, czyli 2^n nazywamy *n-tym poziomem* drzewa \mathbb{T} i oznaczamy symbolem \mathbb{T}_n . Postrzegając elementy drzewa jako ciągi zer i jedynek, poziom n składa się ze wszystkich ciągów o tej samej długości $[i_1, i_2, \dots, i_n]$, gdzie $i_k \in \{0, 1\}$. Poziom 0 składa się z funkcji pustej \emptyset , czyli ciągu pustego $[\]$. Poziom 1 składa się dwóch ciągów: $[0]$ i $[1]$.

3.2 Podstawowe definicje

Metazbiór to jest zbiór, który nie jest w pełni określony, ale — potencjalnie — może zostać na różne sposoby skonkretyzowany. W zależności od pewnych warunków może przyjmować różne konkretne reprezentacje, które są zwykłymi zbiorami. Sformułowanie „w zależności od pewnych warunków” będzie sformalizowane jako interpretacja metazbioru określona przez gałąź w drzewie binarnym.

Z drugiej strony, na metazbiór można patrzeć jak na zwykły zbiór, którego elementy należą do niego w różnych stopniach, nie tylko 0 (nie należy) lub 1 (należy), jak to ma miejsce w klasycznej teorii mnogości. Możliwe stopnie przynależności nie będą określone numerycznie, jak w teorii zbiorów rozmytych, lecz będą reprezentowane przez węzły drzewa binarnego.

3.2.1 Definicja metazbioru

Elementami zwykłych zbiorów są zbiory. Podobnie, elementami metazbiorów będą meta zbiory. Jednak bycie elementem metazbioru oznacza coś więcej niż w przypadku zwykłych zbiorów, ponieważ musi uwzględniać stopień częściowej przynależności elementu do metazbioru. W związku z tym, rzeczywistymi elementami metazbiorów — jeśli spojrzeć na nie jak na zwykłe zbiory — będą pary uporządkowane. Pierwszym elementem takiej pary będzie metazbiór, a drugim — wierzchołek drzewa binarnego (warunek), który będzie określał stopień należenia elementu.

DEFINICJA 3.2.1. *Metazbiór* to zbiór pusty albo zbiór postaci:

$$\tau = \{ \langle \sigma, p \rangle : \sigma \text{ to metazbiór, } p \in \mathbb{T} \} .$$

\mathbb{T} jest drzewem binarnym a $\langle \cdot, \cdot \rangle$ oznacza parę uporządkowaną.

Formalnie, jest to definicja przez indukcję po relacji należenia \in , która jest ufundowana (por. tw. 4.1 w [17, Ch. III, §4]) na mocy Aksjomatu Regularności w Teorii Mnogości Zermelo-Fraenkla (ZFC). Uzasadnienie dla tego rodzaju definicji jest podane za definicją \mathbb{P} -nazwy 2.5 w [17, Ch. VII, §2]. Metazbiór to właśnie szczególny przypadek \mathbb{P} -nazwy z forsingu w ZFC.

Dla zachowania należytej precyzji w zasadzie należałoby napisać:

$$\tau = \{ \langle \sigma_i, p_{ij} \rangle : \sigma_i \text{ to metazbiór, } p_{ij} \in \mathbb{T}, i \in I, j \in J_i \} , \quad (3.2.1)$$

gdzie I , oraz J_i dla $i \in I$, są jakimiś zbiorami indeksów, jednak ten zapis jest mało czytelny, dlatego w dalszym ciągu będziemy stosować pierwszą, uproszczoną formę.

DEFINICJA 3.2.2. Dla metazbioru τ , jeśli $\langle \sigma, p \rangle \in \tau$, to metazbiór σ będzie nazywany *elementem potencjalnym* metazbioru τ .

Pierwszy element σ pary wchodzącej w skład metazbioru τ jest nazywany potencjalnym dlatego, że jego przynależność jako metazbioru σ do samego metazbioru τ jest uwarunkowana (inaczej mówiąc wartościowana) drugim elementem pary, czyli węzłem drzewa \mathbb{T} (warunkiem). Co oznacza owo uwarunkowanie dowiemy się w sekcji 3.3, przy okazji omawiania interpretacji.

Metazbiory oznaczamy małymi literami greckimi: τ, σ , itd. Metazbiór pusty oznaczamy tak, jak zbiór pusty: \emptyset . Klasę wszystkich metazbiorów oznaczamy symbolem \mathfrak{M} .

A więc faktycznymi elementami metazbioru są pary uporządkowane, natomiast jego elementami potencjalnymi są inne metazbiory, będące pierwszymi składnikami tych par. Na metazbiór można patrzeć jak na zwykły zbiór, którego elementy zostały opatrzone etykietkami. Te etykietki to warunki, czyli węzły drzewa \mathbb{T} . Przy tym każdy element potencjalny może występować z wieloma różnymi etykietkami jednocześnie, tworząc w ten sposób wiele różnych składników (par) metazbioru. W związku z tym, zamiast notacji $\langle \sigma, p \rangle$ dla oznaczenia par wchodzących w skład metazbioru będziemy czasem używać skróconej formy: σ^p . Stosując to uproszczenie możemy wyliczyć elementy przykładowego metazbioru:

$$\tau = \{ \sigma^p, \sigma^q, \mu^p, \mu^r \} , \quad (3.2.2)$$

$$= \{ \langle \sigma, p \rangle, \langle \sigma, q \rangle, \langle \mu, p \rangle, \langle \mu, r \rangle \} . \quad (3.2.3)$$

Metazbiór jest relacją, czyli podzbiorem iloczynu kartezjańskiego, pomiędzy swoimi elementami potencjalnymi a drzewem binarnym \mathbb{T} . Z reguły relacja ta nie jest funkcją, gdyż każdy element potencjalny może występować w parach z wieloma różnymi warunkami. W związku z tym możemy wprowadzić następujące definicje.

DEFINICJA 3.2.3. *Dziedziną* metazbioru τ nazywamy zbiór jego potencjalnych elementów:

$$\text{dom}(\tau) = \{ \sigma : \langle \sigma, p \rangle \in \tau \} .$$

DEFINICJA 3.2.4. *Przeciwdziedziną* metazbioru τ nazywamy zbiór:

$$\text{ran}(\tau) = \{ p : \langle \sigma, p \rangle \in \tau \} .$$

Inaczej mówiąc, dziedzina metazbioru to dziedzina relacji, którą ten metazbiór jest. W świetle powyższej definicji

$$\tau \subset \text{dom}(\tau) \times \text{ran}(\tau) \subset \text{dom}(\tau) \times \mathbb{T} . \quad (3.2.4)$$

DEFINICJA 3.2.5. Dla metazbiorów τ i σ , zbiór:

$$\tau[\sigma] = \{ p \in \mathbb{T} : \langle \sigma, p \rangle \in \tau \}$$

nazywamy *obrazem* elementu potencjalnego σ w metazbiorze τ .

Obraz elementu potencjalnego jest zbiorem warunków, czyli węzłów drzewa \mathbb{T} , który określa jego stopień należenia do metazbioru τ (aczkolwiek nie jest to jeszcze formalna definicja relacji należenia dla metazbiorów). Zachodzą następujące związki:

$$\text{ran}(\tau) = \bigcup_{\sigma \in \text{dom}(\tau)} \tau[\sigma] , \quad (3.2.5)$$

$$\tau = \bigcup_{\sigma \in \text{dom}(\tau)} \{ \sigma \} \times \tau[\sigma] . \quad (3.2.6)$$

PRZYKŁAD 3.2.1: Najprostszym metazbiorem jest zbiór pusty: \emptyset . Może on być elementem potencjalnym innego metazbioru:

$$\begin{aligned} \tau &= \{ \langle \emptyset, p \rangle \} , & \tau[\emptyset] &= \{ p \} , & \text{dom}(\tau) &= \{ \emptyset \} , & \text{ran}(\tau) &= \{ p \} , \\ \sigma &= \{ \langle \emptyset, p \rangle, \langle \emptyset, q \rangle \} , & \sigma[\emptyset] &= \{ p, q \} , & \text{dom}(\sigma) &= \{ \emptyset \} , & \text{ran}(\sigma) &= \{ p, q \} . \end{aligned}$$

W pierwszym przypadku stopień należenia \emptyset do τ jest jednoelementowym podzbiorem \mathbb{T} , a w drugim — dwuelementowym (zakładając, że $p \neq q$). Jeśli przy tym $p \top q$, to element silniejszy nie wnosi żadnej dodatkowej informacji i o stopniu przynależności decyduje element słabszy, co się stanie jasne dalej (lemat 4.1.1). Jeśli natomiast $p \perp q$ to oba elementy wnoszą niezależną informację co do sposobu należenia \emptyset do σ .

3.2.2 Ważne klasy metazbiorów

W klasie wszystkich wszystkich metazbiorów \mathfrak{M} wyróżnimy teraz podklasy złożone z metazbiorów posiadających pewne szczególne właściwości. Metazbiory kanoniczne są odpowiednikami zwykłych zbiorów, gdyż relacja należenia dla ich elementów zachodzi zawsze w stopniu $\mathbb{1}$. Z kolei meta zbiory dziedzicznie skończone są tymi, które dają się reprezentować w komputerach, i z tego powodu stanowią podstawę dla wszelkiego rodzaju maszynowych zastosowań.

DEFINICJA 3.2.6. *Metazbiór kanoniczny* to metazbiór pusty \emptyset lub taki metazbiór, w którym pierwsze elementy par są również metazbiorami kanonicznymi, a drugi element jest zawsze korzeniem drzewa \mathbb{T} :

$$\check{\tau} = \{ \langle \check{\sigma}, \mathbb{1} \rangle : \check{\sigma} \text{ jest metazbiorem kanonicznym} \}$$

Metazbiory kanoniczne oznaczamy małymi literami greckimi, czasem dodatkowo ozdobionymi symbolem `''` (`\check`). Klasę metazbiorów kanonicznych oznaczamy symbolem \mathfrak{M}^c .

Porównując budowę metazbiorów kanonicznych i zwykłych zbiorów widzimy, że można ustanowić między nimi standardowy izomorfizm, który opatruje każdy element zwykłego zbioru $x \in X$ etykietką $\mathbb{1}$, tworząc w ten sposób parę uporządkowaną będącą elementem metazbioru. Do elementów zbiorów, które stają się pierwszymi członami par, należy zastosować tę procedurę rekurencyjnie. Symbol `''` można zatem traktować jako jednoargumentowy operator, który ze swojego argumentu, będącego zwykłym zbiorem, tworzy metazbiór będący jego kanoniczną wersją.

DEFINICJA 3.2.7. *Izomorfizm kanoniczny* $\check{\cdot} : \mathbf{V} \mapsto \mathfrak{M}^c$ przyporządkowuje każdemu zbiorowi jego kanoniczny odpowiednik w następujący sposób:

$$\begin{aligned} \check{\cdot} : \emptyset &\mapsto \check{\emptyset}, && \text{(oczywiście, } \check{\emptyset} = \emptyset) \\ \check{\cdot} : \{x_i\} &\mapsto \{ \langle \check{x}_i, \mathbb{1} \rangle \}. && \text{(dla } x \in \mathbf{V}, x = \{x_i\}) \end{aligned}$$

Dziedzina izomorfizmu kanonicznego jest klasa wszystkich zbiorów \mathbf{V} , a przeciwdziedzina — klasa wszystkich metazbiorów kanonicznych \mathfrak{M}^c .

PRZYKŁAD 3.2.2: Tak wyglądają początkowe liczby porządkowe i ich kanoniczne odpowiedniki:

$$\begin{aligned} \check{0} &= \emptyset, && \text{(zero)} \\ \check{1} &= \overline{\{0\}} = \{ \langle \check{0}, \mathbb{1} \rangle \} = \{ \langle \emptyset, \mathbb{1} \rangle \}, && \text{(jeden)} \\ \check{2} &= \overline{\{0, 1\}} = \{ \langle \check{0}, \mathbb{1} \rangle, \langle \check{1}, \mathbb{1} \rangle \} = \{ \langle \check{0}, \mathbb{1} \rangle, \langle \overline{\{0\}}, \mathbb{1} \rangle \} \\ &= \{ \langle \emptyset, \mathbb{1} \rangle, \langle \{ \langle \emptyset, \mathbb{1} \rangle \}, \mathbb{1} \rangle \}. && \text{(dwa)} \end{aligned}$$

Symbole $\check{0}$, $\check{1}$ i $\check{2}$ oznaczają tu definiowane metazbiory kanoniczne odpowiadające kolejnym liczbom porządkowym (wyjątkowo, dla zwiększenia przejrzystości, nie użyliśmy greckich liter).

Jak wiadomo, w komputerach można reprezentować wyłącznie obiekty skończone. Z tego powodu, wyodrębnimy podklasę metazbiorów, które takie są. Oczywiście, sama skończoność metazbioru jako zbioru (par) może nie zapewnić komputerowej reprezentowalności, gdyż elementy potencjalne również muszą być skończenie reprezentowalne.

DEFINICJA 3.2.8. Metazbiór τ jest *dziedzicznie skończony*, jeśli jego dziedzina i przeciwdziedzina są skończone, oraz każdy potencjalny element jest metazbiorem dziedzicznie skończonym.

Klasę metazbiorów dziedzicznie skończonych oznaczamy symbolem $\mathfrak{M}\check{\mathfrak{F}}$. Zatem:

$$\tau \in \mathfrak{M}\check{\mathfrak{F}} \text{ gdy: } |\text{dom}(\tau)| < \aleph_0 \wedge |\text{ran}(\tau)| < \aleph_0 \wedge \forall_{\sigma \in \text{dom}(\tau)} \sigma \in \mathfrak{M}\check{\mathfrak{F}} . \quad (3.2.7)$$

Jeśli τ jest dziedzicznie skończony, to jest też skończony jako zwykły zbiór:

$$|\tau| \leq |\text{dom}(\tau)| \cdot |\text{ran}(\tau)| \text{ bo } \tau \subset \text{dom}(\tau) \times \text{ran}(\tau) . \quad (3.2.8)$$

Ponadto, obraz $\tau[\sigma]$ każdego potencjalnego elementu σ w metazbiorze τ jest reprezentowany przez zbiór skończony:

$$\forall_{\sigma \in \text{dom}(\tau)} |\tau[\sigma]| < \aleph_0 \text{ bo } \tau[\sigma] \subset \text{ran}(\tau) . \quad (3.2.9)$$

3.3 Interpretacje metazbiorów

Interpretacja metazbioru to zwykły zbiór. Jest on spojrzeniem na metazbiór z pewnej perspektywy. Każdy metazbiór może mieć wiele różnych interpretacji. Właściwości interpretacji metazbioru będą definiować właściwości samego metazbioru.

3.3.1 Definicja interpretacji

Interpretacja metazbioru jest zadana przez gałąź w drzewie binarnym \mathbb{T} .

DEFINICJA 3.3.1. Jeśli τ jest metazbiorem, to jego *interpretacja* zadana przez gałąź $\mathcal{C} \subset \mathbb{T}$ jest zbiorem:

$$\text{int}(\tau, \mathcal{C}) = \{ \text{int}(\sigma, \mathcal{C}) : \langle \sigma, p \rangle \in \tau \wedge p \in \mathcal{C} \} . \quad (3.3.1)$$

Interpretację metazbioru τ zadaną przez gałąź \mathcal{C} przeważnie będziemy oznaczać skróconą formą $\tau_{\mathcal{C}}$:

$$\tau_{\mathcal{C}} = \{ \sigma_{\mathcal{C}} : \sigma^p \in \tau \wedge p \in \mathcal{C} \} . \quad (3.3.2)$$

Jak widać, interpretacja powstaje z metazbioru poprzez odrzucenie etykietek (drugich elementów par, warunków) z elementów potencjalnych. W jej skład wchodzi interpretacje tych elementów potencjalnych, których etykiety należą do gałęzi \mathcal{C} . Procedurę tę stosujemy rekurencyjnie.

PRZYKŁAD 3.3.1: Zinterpretujemy metazbiory z przykładu 3.2.2. Dla dowolnej gałęzi \mathcal{C} (bo $\mathbb{1} \in \mathcal{C}$ zawsze):

$$\begin{aligned}\emptyset_{\mathcal{C}} &= \emptyset, \\ \check{1}_{\mathcal{C}} &= \{\langle \emptyset, \mathbb{1} \rangle\}_{\mathcal{C}} = \{\emptyset\} = 1, \\ \check{2}_{\mathcal{C}} &= \{\langle \emptyset, \mathbb{1} \rangle, \langle \{\langle \emptyset, \mathbb{1} \rangle\}, \mathbb{1} \rangle\}_{\mathcal{C}} = \{\emptyset, \{\emptyset\}\} = \{0, 1\} = 2.\end{aligned}$$

Widzimy, że interpretacja metazbioru kanonicznego jest zawsze taka sama niezależna od gałęzi \mathcal{C} (por. fakt 3.3.3). Interpretacje metazbiorów, które nie są kanoniczne mogą być różne dla różnych gałęzi.

PRZYKŁAD 3.3.2: Niech $p, q \in \mathbb{T}$ i $p \perp q$, na przykład: $p = [01]$, $q = [00]$. Niech $\sigma = \{\langle \check{1}, p \rangle, \langle \check{2}, q \rangle\}$ (oznaczenia z przykł. 3.2.2). Jeśli \mathcal{C} jest gałęzią, to

$$\begin{aligned}p \in \mathcal{C} &\rightarrow \sigma_{\mathcal{C}} = \{1\}, && \text{(bo } q \notin \mathcal{C}\text{)} \\ q \in \mathcal{C} &\rightarrow \sigma_{\mathcal{C}} = \{2\}, && \text{(bo } p \notin \mathcal{C}\text{)} \\ p \notin \mathcal{C} \wedge q \notin \mathcal{C} &\rightarrow \sigma_{\mathcal{C}} = 0 = \emptyset. && \text{(wtedy } [1] \in \mathcal{C}\text{)}\end{aligned}$$

Powyższe przypadki wzajemnie się wykluczają, bo $p \perp q \perp [1]$ (nie mogą leżeć na jednej gałęzi, bo nie są porównywalne). A więc zależnie od wyboru gałęzi \mathcal{C} mamy różne interpretacje metazbioru σ .

Powyższy przykład pokazuje, że metazbiór może reprezentować wiele różnych ostrych zbiorów jednocześnie.

3.3.2 Podstawowe fakty

Przedstawimy teraz najprostsze właściwości metazbiorów i interpretacji niezbędne dla dalszej prezentacji. Szersze ujęcie tego tematu można znaleźć w pracy [31]. Proste dowody przytoczonych faktów wynikają bezpośrednio z definicji metazbioru i interpretacji.

Po pierwsze odnotujemy, że złożenie izomorfizmu kanonicznego i interpretacji jest identycznością na klasie wszystkich zbiorów \mathbf{V} .

FAKT 3.3.1. Jeśli $x \in \mathbf{V}$ jest zbiorem, a \check{x} jego kanonicznym odpowiednikiem, danym przez izomorfizm kanoniczny, to dla dowolnej gałęzi \mathcal{C} jest $\check{x}_{\mathcal{C}} = x$.

Z powyższego wynika, że różne metazbiory kanoniczne mają różne interpretacje, co w ogólnym przypadku nie jest prawdą.

FAKT 3.3.2. Jeśli $\tau, \sigma \in \mathfrak{M}^c$ oraz $\tau \neq \sigma$, to dla każdej gałęzi \mathcal{C} jest $\tau_{\mathcal{C}} \neq \sigma_{\mathcal{C}}$.

Wszystkie interpretacje metazbioru kanonicznego są identyczne. Oczywiście, w ogólnym przypadku nie jest to prawdą, co pokazuje przykład 3.3.2.

FAKT 3.3.3. Jeśli \mathcal{C}' i \mathcal{C}'' są różnymi gałęziami a $\check{\tau} \in \mathfrak{M}^c$, to $\check{\tau}_{\mathcal{C}'} = \check{\tau}_{\mathcal{C}''}$.

Każdy element x interpretacji $\tau_{\mathcal{C}}$ metazbioru τ jest interpretacją jakiegoś metazbioru σ z dziedziny $\text{dom}(\tau)$.

FAKT 3.3.4. Niech τ będzie niepustym metazbiorem, a \mathcal{C} niech będzie gałęzią. Jeśli $x \in \tau_{\mathcal{C}}$, to istnieje para $\langle \sigma, p \rangle \in \tau$ taka, że $p \in \mathcal{C}$ oraz $\sigma_{\mathcal{C}} = x$.

Dowód. Z definicji metazbioru $\tau_{\mathcal{C}} = \{\sigma_{\mathcal{C}} : \exists p \in \mathcal{C} \langle \sigma, p \rangle \in \tau\}$. Zatem jeśli $x \in \tau_{\mathcal{C}}$, to $x = \sigma_{\mathcal{C}}$ dla pewnego $\sigma \in \text{dom}(\tau)$ takiego, że $\langle \sigma, p \rangle \in \tau$ i $p \in \mathcal{C}$. \square

W ogólnym przypadku takich par może być wiele. Oczywiście jest, że mogą się one różnić drugim elementem — warunkiem p . Mniej oczywiste jest, że mogą istnieć różne pary $\langle \eta, p \rangle \in \tau$ i $\langle \sigma, p \rangle \in \tau$, gdzie $\sigma \neq \eta$, jednak takie, że $\sigma_{\mathcal{C}} = \eta_{\mathcal{C}} = x$.

PRZYKŁAD 3.3.3: Niech $p \in \mathbb{T}$ i niech $\sigma = \{\langle \check{0}, p \rangle\}$, a $\eta = \{\langle \check{0}, p \cdot 0 \rangle, \langle \check{0}, p \cdot 1 \rangle\}$. Niech ponadto $\tau = \{\sigma^{\mathbb{1}}, \eta^{\mathbb{1}}\}$, a \mathcal{C} niech będzie gałęzią. Widzimy, że

$$\begin{aligned} p \in \mathcal{C} &\rightarrow \sigma_{\mathcal{C}} = \eta_{\mathcal{C}} = \{\emptyset\} = 1, \\ p \notin \mathcal{C} &\rightarrow \sigma_{\mathcal{C}} = \eta_{\mathcal{C}} = \emptyset = 0. \end{aligned}$$

Zatem ponieważ $\tau_{\mathcal{C}} = \{\sigma_{\mathcal{C}}, \eta_{\mathcal{C}}\}$, to

$$\begin{aligned} p \in \mathcal{C} &\rightarrow \tau_{\mathcal{C}} = \{1\}, \\ p \notin \mathcal{C} &\rightarrow \tau_{\mathcal{C}} = \{0\}. \end{aligned}$$

Jeśli jednak elementami dziedziny metazbioru τ są wyłącznie meta zbiory kanoniczne, to odwrotna interpretacja jest jednoznaczna, co pokazuje kolejny fakt.

FAKT 3.3.5. Niech τ będzie metazbiorem takim, że $\text{dom}(\tau) \subset \mathfrak{M}^c$, a \mathcal{C} niech będzie gałęzią. Jeśli $x \in \tau_{\mathcal{C}}$, to istnieje dokładnie jeden metazbiór kanoniczny $\check{\sigma} \in \text{dom}(\tau)$ taki, że $x = \check{\sigma}_{\mathcal{C}}$. Jest on zadany przez kanoniczny izomorfizm: $\check{x} = \check{\sigma}$.

Dowód. Istnienie $\check{\sigma} \in \text{dom}(\tau)$ takiego, że $\check{\sigma}_{\mathcal{C}} = x$, wynika z faktu 3.3.4, jednoznaczność z faktu 3.3.2. \square

Rozdział 4

Działania na metazbiorach

W tym rozdziale zdefiniujemy podstawowe relacje dla metazbiorów, czyli należenie, równość i zawieranie. W przeciwieństwie do klasycznych, ostrych zbiorów, a podobnie do zbiorów rozmytych, relacje te będą mogły zachodzić w różnych stopniach, nie tylko 0 lub 1.

To, że należenie będzie uwzględniać różne poziomy było widoczne już w poprzednim rozdziale i wynikało z samej budowy metazbioru. Jeśli chodzi jednak o pozostałe relacje, to do ich zdefiniowania niezbędny będzie pewien ogólny mechanizm, który wprowadzimy najpierw.

Niniejsza praca koncentruje się wokół zastosowań komputerowych teorii metazbiorów, w związku z tym prezentacja wyników teoretycznych została ograniczona do niezbędnego minimum. Mechanizmy, których fragmenty tu zaprezentujemy (por. też [29]), stanowią istotny fundament teoretyczny dla koncepcji metazbiorów. Pozwolą w przyszłości między innymi na zdefiniowanie wielowartościowej logiki dla języka ich opisu. Umożliwiają również reprezentowanie przy pomocy metazbiorów nie tylko zwykłych zbiorów rozmytych typu 1 i 2 (por. rozdz. 6), ale również intuicjonistycznych zbiorów rozmytych ([30]), które charakteryzują się tym, że ich elementy, oprócz stopni przynależności i nieprzynależności, posiadają również stopień niepewności.

Definicje podstawowych relacji przedstawione w tym rozdziale mają przede wszystkim znaczenie teoretyczne. Do praktycznych zastosowań komputerowych lepiej będą nadawały się sformułowania tych samych relacji wprowadzone w części 5 dla pewnej, istotnej dla zastosowań, klasy metazbiorów. Oczywiście, będą one sobie równoważne na tej klasie.

4.1 Wymuszanie

Relacja wymuszania, którą wprowadzimy poniżej, jest podstawowym narzędziem służącym do definiowania właściwości metazbiorów. Będzie ona wyrażać związek pomiędzy warunkiem a zdaniem,¹ który określa warunkową prawdziwość tego zdania, jego poziom prawdziwości. Warunek to element drzewa binarnego \mathbb{T} , natomiast zdanie będzie formułą klasycznej teorii zbiorów, w której

¹Definiowanie relacji pomiędzy elementami drzewa binarnego a zdaniami w jakimś języku może się wydawać dziwne, jednak można tę ideę sformalizować implementując logikę w teorii mnogości w sposób analogiczny do gödelizacji języka w arytmetyce.

za zmienne wolne są podstawione termy języka metazbiorów, czyli jakieś metazbiory. Takie zdanie może być prawdziwe lub nie (w sensie klasycznej ZFC), w zależności od interpretacji metazbiorów, o których ono mówi. Każdy warunek wyznacza pewną rodzinę interpretacji – są one określone przez zawierające go gałęzie. Jeśli w każdej interpretacji zawierającej dany warunek, zdanie ma tę samą wartość logiczną (prawda lub fałsz) to możemy mówić o warunkowej prawdziwości (fałszywości) tego zdania, zdeterminowanej przez ten właśnie warunek.

4.1.1 Definicja wymuszania

Niech Φ będzie formułą zbudowaną ze zmiennych x^1, x^2, \dots , symbolu stałej: \emptyset , symboli relacji: $\in, =, \subset$, spójników logicznych: $\wedge, \vee, \neg, \rightarrow$, kwantyfikatorów: \forall, \exists oraz nawiasów. Jeśli za każdą zmienną wolną x^i ($i = 1 \dots n$) tej formuły podstawimy jakiś metazbiór ν^i , a zakres każdego kwantyfikatora ograniczymy do klasy metazbiorów \mathfrak{M} , to otrzymamy zdanie $\Phi(\nu^1, \dots, \nu^n)$, wyrażające jakąś właściwość metazbiorów ν^1, \dots, ν^n . Przez Φ_C będziemy oznaczać interpretację tego zdania, czyli zdanie $\Phi(\nu_C^1, \dots, \nu_C^n)$. Powstało ono poprzez podstawienie za zmienne wolne formuły Φ interpretacji metazbiorów ν_C^i i ograniczenie występujących w niej kwantyfikatorów do uniwersum wszystkich zbiorów \mathbf{V} . Stała \emptyset zarówno w Φ jak i w Φ_C oznacza zbiór pusty, który jest w obu przypadkach (jako metazbiór i jako zwykły zbiór) tym samym zbiorem.

DEFINICJA 4.1.1. Niech x^1, x^2, \dots, x^n będą wszystkimi zmiennymi wolnymi w formule Φ , a $\nu^1, \nu^2, \dots, \nu^n \in \mathfrak{M}$ niech będą metazbiorami. Powiemy, że warunek $p \in \mathbb{T}$ wymusza zdanie $\Phi(\nu^1, \nu^2, \dots, \nu^n)$, gdy dla każdej gałęzi C w drzewie \mathbb{T} zawierającej warunek p , zachodzi $\Phi(\nu_C^1, \nu_C^2, \dots, \nu_C^n)$. Relację wymuszania oznaczamy symbolem \Vdash . Zatem:

$$p \Vdash \Phi(\nu^1, \nu^2, \dots, \nu^n) \text{ gdy dla każdej gałęzi } C: p \in C \rightarrow \Phi(\nu_C^1, \nu_C^2, \dots, \nu_C^n) .$$

Zwykle będziemy pomijać listę zmiennych stosując skrócony zapis: $p \Vdash \Phi$, mając wtedy na uwadze, że dla każdej gałęzi C zawierającej p jest prawdziwe zdanie Φ_C .

Będziemy używać skrótu $p \not\Vdash \Phi$ dla zaprzeczenia: $\neg(p \Vdash \Phi)$. W takiej sytuacji nie dla każdej gałęzi C zawierającej p zachodzi Φ_C , jednak mogą takie gałęzie istnieć. Z kolei $\mathbb{1} \Vdash \Phi$ będzie oznaczać $\mathbb{1} \Vdash \Phi$. Jeśli $\mathbb{1}$ wymusza Φ , to znaczy, że Φ zachodzi zawsze, co usprawiedliwia skróconą notację. Dodatkowo symbol \notin w formule $\mu \notin \tau$ będzie skrótem dla $\neg(\mu \in \tau)$, $\mu \neq \tau$ będzie skrótem dla $\neg(\mu = \tau)$, a $\mu \not\subset \tau$ będzie skrótem dla $\neg(\mu \subset \tau)$.

Następujący lemat pokazuje, że wymuszanie jest propagowane w dół gałęzi. Jeśli warunek p wymusza Φ , to warunki silniejsze tym bardziej. Nie jest jednak na odwrót, o czym świadczy przykład 4.1.2. Zatem warunki silniejsze niosą istotnie więcej informacji.

LEMAT 4.1.1. Dla $p, q \in \mathbb{T}$ i zdania Φ zachodzi implikacja

$$p \Vdash \Phi \wedge q \leq p \rightarrow q \Vdash \Phi .$$

Dowód. $q \Vdash \Phi$ oznacza, że dla każdej gałęzi $C \ni q$ jest Φ_C . Ale każda gałąź zawierająca q zawiera również p , a z założenia dla każdej takiej gałęzi prawdziwe jest Φ_C . \square

Z drugiej strony, jeśli $R \subset \mathbb{T}$ zawiera skończony antyłańcuch maksymalny poniżej $p \in \mathbb{T}$, to warunki ze zbioru R silniejsze od p indukują wymuszanie zdania Φ przez p .

LEMAT 4.1.2. Jeśli $p \in \mathbb{T}$, $A \subset \mathbb{T}$ jest skończonym antyłańcuchem maksymalnym poniżej p , a Φ jest zdaniem, to prawdziwa jest implikacja

$$\bigvee_{q \in A} q \Vdash \Phi \quad \rightarrow \quad p \Vdash \Phi .$$

Dowód. Skoro A jest skończonym antyłańcuchem maksymalnym, którego każdy element leży poniżej p , to każda gałąź przechodząca przez p musi zawierać jakiś element $q \in A$. Rzeczywiście, niech $n \in \mathbb{N}$ będzie takie, że wszystkie elementy antyłańcucha A leżą na poziomach o numerach mniejszych niż n (do \mathbb{T}_{n-1} włącznie). Takie n istnieje, bo A jest skończony. W takim razie dla każdego $r \in \mathbb{T}_n$, które leży poniżej p , istnieje $s \in A$ takie, że $r \leq s$, bo gdyby tak nie było, to by istniało $r \in \mathbb{T}_n$ leżące poniżej p i nieporównywalne z żadnym elementem z A . Wtedy $A \cup \{r\}$ byłby antyłańcuchem, co przeczy maksymalności A . Ponieważ każdy element \mathbb{T}_n silniejszy od p ma nad sobą element z A , to każda gałąź \mathcal{C} przechodząca przez p musi zawierać też jakiś $q \in A$. Z założenia mamy $q \Vdash \Phi$. Wynika z tego, że w każdej interpretacji zadanej przez $\mathcal{C} \ni p$ jest $\Phi_{\mathcal{C}}$, a to oznacza, że $p \Vdash \Phi$. \square

Na warunek p , który wymusza jakieś zdanie, można patrzeć jak na stopień prawdziwości tego zdania. Warunek słabszy to większy stopień prawdziwości, a $\mathbb{1}$ to pewność. Zdanie wymuszane przez $\mathbb{1}$ jest prawdziwe w każdej interpretacji.

4.1.2 Przykłady

PRZYKŁAD 4.1.1: Niech $\tau = \{\langle \emptyset, \mathbb{1} \rangle\}$ oraz $\sigma = \{\langle \emptyset, [0] \rangle, \langle \emptyset, [1] \rangle\}$. Mamy $\mathbb{1} \Vdash \tau = \sigma$, bo zbiór warunków $\sigma[\emptyset] = \{[0], [1]\}$ jest antyłańcuchem maksymalnym, i dla każdej interpretacji \mathcal{C} mamy $\tau_{\mathcal{C}} = \sigma_{\mathcal{C}} = \{\emptyset\} = 1$.

Weźmy dowolne p zamiast $\mathbb{1}$: $\tau = \{\langle \emptyset, p \rangle\}$ i $\sigma = \{\langle \emptyset, p \cdot 0 \rangle, \langle \emptyset, p \cdot 1 \rangle\}$. Tym razem $p \Vdash \tau = \sigma$ i dla interpretacji \mathcal{C} zawierającej p mamy jak poprzednio $\tau_{\mathcal{C}} = \sigma_{\mathcal{C}} = \{\emptyset\}$. Jeśli jednak $p \notin \mathcal{C}$ to $\tau_{\mathcal{C}} = \emptyset = \sigma_{\mathcal{C}}$, więc również zachodzi $\Vdash \tau = \sigma$.

Zatem jeśli w obrazie $\tau[\eta]$ (dla jakichś τ, η) zastąpimy warunek p skończonym antyłańcuchem maksymalnym poniżej p (lub na odwrót), to otrzymamy metazbiór równoważny wyjściowemu, o takich samych właściwościach.

PRZYKŁAD 4.1.2: Niech $p \in \mathbb{T}$ oraz niech μ będzie takie:

$$\mu = \{ \langle \check{0}, p \cdot 0 \rangle, \langle \check{1}, p \cdot 1 \rangle \} = \{ \langle \emptyset, p \cdot 0 \rangle, \langle \{ \emptyset^1 \}, p \cdot 1 \rangle \} .$$

Mamy:

$$\begin{aligned} p \cdot 0 \Vdash \mu = \check{1} \quad , \quad \text{bo} \quad p \cdot 0 \Vdash \mu = \{ \check{0}^{p \cdot 0} \} \quad , \\ p \cdot 1 \Vdash \mu \neq \check{1} \quad , \quad \text{bo} \quad p \cdot 1 \Vdash \mu = \{ \check{1}^{p \cdot 1} \} \quad . \end{aligned}$$

Dla gałęzi \mathcal{C} zawierającej $p \cdot 0$ będzie $\mu_{\mathcal{C}} = \{\emptyset\} = 1$. Natomiast dla $\mathcal{C} \ni p \cdot 1$ będzie $\mu_{\mathcal{C}} = \{1\} = \{\{\emptyset\}\}$.

Widzimy, że nieporównywalne warunki mogą wymuszać sprzeczne formuły.

PRZYKŁAD 4.1.3: Niech $p, q \in \mathbb{T}$ będą nieporównywalne, a $\tau = \{\langle \emptyset, p \rangle\}$ oraz $\sigma = \{\langle \emptyset, q \rangle\}$. Mamy:

$$\begin{aligned} p \Vdash \tau \neq \sigma, & \text{ bo } p \in \mathcal{C} \rightarrow q \notin \mathcal{C} \rightarrow \tau_{\mathcal{C}} = \{\emptyset\} \wedge \sigma_{\mathcal{C}} = \emptyset \rightarrow \tau_{\mathcal{C}} \neq \sigma_{\mathcal{C}}, \\ \text{i: } q \Vdash \tau \neq \sigma, & \text{ bo } q \in \mathcal{C} \rightarrow p \notin \mathcal{C} \rightarrow \tau_{\mathcal{C}} = \emptyset \wedge \sigma_{\mathcal{C}} = \{\emptyset\} \rightarrow \tau_{\mathcal{C}} \neq \sigma_{\mathcal{C}}, \\ \text{ale: } \mathbb{1} \not\Vdash \tau \neq \sigma, & \text{ bo } p \notin \mathcal{C} \wedge q \notin \mathcal{C} \rightarrow \tau_{\mathcal{C}} = \emptyset \wedge \sigma_{\mathcal{C}} = \emptyset \rightarrow \tau_{\mathcal{C}} = \sigma_{\mathcal{C}}. \end{aligned}$$

Zatem jedne warunki mogą wymuszać jakąś formułę, inne nie — mogą nawet wymuszać jej zaprzeczenie. Ostatnia teza zachodzi pod warunkiem, że istnieje gałąź \mathcal{C} niezawierająca ani p ani q , co oznacza, że przynajmniej jedno z p, q nie należy do zbioru warunków $\{[0], [1]\}$, które są bezpośrednimi potomkami $\mathbb{1}$ w drzewie \mathbb{T} .

Powyższy przykład pokazał, że $p \not\Vdash \Phi$ nie jest równoważne z $p \Vdash \neg\Phi$.

4.2 Podstawowe relacje

Technika wymuszania pozwoli nam zdefiniować podstawowe relacje dla metazbiorów. W szczególności zdefiniujemy relację należenia dla metazbiorów tak, że będzie ona oddawała sens „częściowego należenia”. Podobnie, relacje równości i zawierania się metazbiorów będą uwzględniać fakt, że metazbiory mogą należeć „częściowo”.

Każdej ze standardowych relacji będzie odpowiadać przeliczalnie wiele symboli relacyjnych aby umożliwić wyrażenie różnych stopni spełnienia każdej relacji. Stopnie będą reprezentowane przez węzły drzewa \mathbb{T} .

Ze względu na przeliczalną liczbę symboli relacyjnych język opisu metazbiorów będzie nieskończony. Będziemy jednak używać klasycznej logiki dwuwartościowej. Właśnie dzięki nieskończonej liczbie symboli opisujących zawile związki pomiędzy metazbiorami będzie to możliwe

4.2.1 Należenie warunkowe

Jeśli potraktować metazbiór jak zwykły zbiór, to jego elementami będą pary uporządkowane. Jednak pary te reprezentują metazbiory. Zatem stwierdzenie „metazbiór μ należy do meta zbioru τ ” ma pewien sens. Takie należenie będziemy nazywać *należeniem warunkowym*, bo jest ono zdeterminowane przez warunki — drugie elementy par definiujących metazbiór.

DEFINICJA 4.2.1. Niech $p \in \mathbb{T}$. Powiemy, że metazbiór μ należy do meta zbioru τ *pod warunkiem* p (ozn. $\mu \epsilon_p \tau$), jeśli $p \Vdash \mu \in \tau$.

Inaczej mówiąc, $\mu \epsilon_p \tau$, jeśli dla każdej gałęzi \mathcal{C} zawierającej p , zachodzi $\mu_{\mathcal{C}} \in \tau_{\mathcal{C}}$. Należenie warunkowe odzwierciedla fakt, że metazbiór μ może w pewnych okolicznościach należeć do meta zbioru τ a w innych nie. Można również takiemu należeniu przyporządkować wartość numeryczną, co zobaczymy w części 6.

Elementów p, q, r, \dots , które warunkują należenie, jest zawsze wiele. Na przykład, jeśli $p \Vdash \mu \in \tau$ i $q \leq p$, to również $q \Vdash \mu \in \tau$ (na mocy lematu 4.1.1). Wydawałoby się więc, że warunkowe należenie nie jest jednoznaczne. Można jednak, np. wybierając elementy maksymalne spośród tych, które warunkują należenie, określić „bazę” warunków, która będzie w jednoznaczny sposób specyfikować stopień częściowego należenia metazbioru do innego.

Szczególnym przypadkiem warunku jest $\mathbb{1}$. Jeśli $\mu \in_{\mathbb{1}} \tau$, czyli $\Vdash \mu \in \tau$, to μ_C należy do τ_C zawsze, w każdej interpretacji \mathcal{C} .

DEFINICJA 4.2.2. Niech μ, τ będą metazbiorami, a $p \in \mathbb{T}$. Powiemy, że μ należy do τ (ozn. $\mu \in \tau$), jeśli $\Vdash \mu \in \tau$.

Zwróćmy uwagę, że nawet jeśli $\mu \in \tau$, to w każdej interpretacji \mathcal{C} , μ_C i τ_C mogą być innymi zbiorami. Symbol \in mówi, że w każdym takim przypadku relacja \in pomiędzy interpretacjami μ_C i τ_C jest mimo to zachowana.

Co to znaczy, że $\neg \mu \in_p \tau$? Z definicji mamy: $p \nVdash \mu \in \tau$ – warunek p nie wymusza $\mu \in \tau$. Może istnieć jednak inny warunek q taki, że $q \Vdash \mu \in \tau$ czyli $\mu \in_q \tau$. Oddaje to sens częściowego należenia bądź częściowego nienależenia μ do τ . Mówiąc bardzo nieformalnie: jeśli jakiś fragment μ (określony przez p) jest poza τ , to inny fragment (określony przez q) może być w τ .

Jeśli p jest korzeniem drzewa \mathbb{T} , to znaczenie $\neg \mu \in_{\mathbb{1}} \tau$, czyli $\neg \mu \in \tau$, jest dane przez $\nVdash \mu \in \tau$. W tym przypadku nie jest prawdą, że w każdej interpretacji \mathcal{C} zachodzi $\mu_C \in \tau_C$. Nic nie stoi jednak na przeszkodzie, by taka interpretacja istniała. W ogólnym przypadku nie wiadomo, czy jest ona wymuszana przez jakiś warunek. Jeśli jednak $\mu, \tau \in \mathfrak{M}\mathfrak{F}$, to można udowodnić ([29]), że wtedy istnieje q takie, że $\mu \in_q \tau$.

Widzimy więc, że $\neg \mu \in_p \tau$ nie wyklucza całkowicie należenia μ do τ nawet, jeśli $p = \mathbb{1}$. To, że $\neg \mu \in \tau$ nie wyklucza $\mu \in_p \tau$ dla pewnego p stwierdza tylko, że μ nie może należeć do τ w stopniu $\mathbb{1}$.

DEFINICJA 4.2.3. Dla $p \in \mathbb{T}$ powiemy, że μ nie należy do τ (ozn. $\mu \notin \tau$), jeśli zachodzi $\nVdash \mu \notin \tau$.

Inaczej mówiąc $\mu \notin \tau$, jeśli dla każdej gałęzi \mathcal{C} zachodzi $\mu_C \notin \tau_C$.

DEFINICJA 4.2.4. Dla $p \in \mathbb{T}$ powiemy, że μ nie należy do τ pod warunkiem p (ozn. $\mu \notin_p \tau$), jeśli zachodzi $p \nVdash \mu \notin \tau$.

Dla $p \perp q$ możliwe jest, że $\mu \notin_p \tau$ i jednocześnie $\mu \in_q \tau$. Ale dla żadnego p nie jest prawdą, że $\mu \notin_p \tau \wedge \mu \in_p \tau$. Można udowodnić ([29]), że dla $\sigma, \tau \in \mathfrak{M}\mathfrak{F}$ zachodzi dokładnie jedna z trzech możliwości: $\sigma \in \tau$ lub $\sigma \notin \tau$ lub $\exists_{p,q} (\sigma \in_p \tau \wedge \sigma \notin_q \tau)$.

Podsumujmy związki pomiędzy \in i \notin w kontekście negacji.

LEMAT 4.2.1. Niech $p, q \in \mathbb{T}$, a μ, τ będą metazbiorami. Jeśli p jest porównywalne z q , to:

$$\begin{aligned} \mu \in_p \tau &\rightarrow \neg (\mu \notin_q \tau) \quad , \\ \mu \notin_p \tau &\rightarrow \neg (\mu \in_q \tau) \quad . \end{aligned}$$

Dowód. Zdanie $\neg \mu \notin_q \tau$ mówi, że istnieje gałąź $\mathcal{C} \ni q$ taka, że $\mu_C \in \tau_C$. Z kolei $\mu \in_p \tau$ mówi, że dla każdej gałęzi $\mathcal{C} \ni p$ jest $\mu_C \in \tau_C$. Jeśli $q \geq p$, to każda gałąź przechodząca przez p zawiera również q . Dla każdej takiej gałęzi mamy $\mu_C \in \tau_C$, więc istnieją gałęzie przechodzące przez q takie, że $\mu_C \in \tau_C$ a stąd $\neg \mu \notin_q \tau$. Jeśli $q \leq p$, to każda gałąź zawierająca q zawiera również p . Skoro zatem dla każdej gałęzi $\mathcal{C} \ni p$ mamy $\mu_C \in \tau_C$, to tym bardziej dla każdej $\mathcal{C} \ni q$ mamy to samo. W tym przypadku nie tylko $\neg \mu \notin_q \tau$ ale nawet $\mu \in_q \tau$. Drugiej implikacji dowodzimy analogicznie. \square

Jeśli przez warunek p przechodzi gałąź \mathcal{C} , dla której $\mu_C \notin \tau_C$, to nie znaczy, że ten związek zachodzi dla każdej takiej gałęzi. Tym bardziej nie musi on zachodzić dla wszystkich gałęzi zawierających warunek słabszy niż p .

LEMAT 4.2.2. Niech $p, q \in \mathbb{T}$. Jeśli $p \leq q$, to:

$$\begin{aligned} \neg(\mu \in_p \tau) &\leftrightarrow \mu \notin_q \tau, \\ \neg(\mu \notin_p \tau) &\leftrightarrow \mu \in_q \tau. \end{aligned}$$

Dowód. Niech $p \in \mathbb{T}$ oraz $\tau = \{\langle \emptyset, \mathbb{1} \rangle\}$ i $\mu = \{\langle \emptyset, p \cdot 0 \rangle\}$. Mamy:

$$\begin{aligned} p \cdot 0 \Vdash \mu \notin \tau, \quad \text{bo dla gałęzi } C_0 \ni p \cdot 0 \text{ jest: } \mu_{C_0} = \{\emptyset\} \text{ i } \tau_{C_0} = \{\emptyset\}, \\ p \cdot 1 \Vdash \mu \in \tau, \quad \text{bo dla gałęzi } C_1 \ni p \cdot 1 \text{ jest: } \mu_{C_1} = \emptyset \text{ i } \tau_{C_1} = \{\emptyset\}. \end{aligned}$$

Zatem $p \not\Vdash \mu \in \tau$ i $p \not\Vdash \mu \notin \tau$, czyli $\neg \mu \in_p \tau \wedge \neg \mu \notin_p \tau$. Jeśli $q \geq p$, to tym bardziej: $\neg \mu \in_p \tau \wedge \neg \mu \notin_q \tau$, jak również $\neg \mu \notin_p \tau \wedge \neg \mu \in_q \tau$. To dowodzi, że żadna z implikacji nie może być prawdziwa. \square

Jeśli $p > q$, to w ogólnym przypadku nie da się rozstrzygnąć czy implikacje z lematu 4.2.2 mogą być prawdziwe czy też nie. Jeśli $p \perp q$, to (nie)należenie pod warunkiem p jest całkowicie niezależne od (nie)należenia pod warunkiem q .

Stopień należenia σ do τ jest zdeterminowany przez zbiór tych $p \in \mathbb{T}$, że $\sigma \in_p \tau$. Podobnie jest ze stopniem nienależenia, który jest zdeterminowany przez zbiór tych $q \in \mathbb{T}$, że $\sigma \notin_q \tau$. Ponieważ jednak może się zdarzyć, że dla pewnych $p \in \mathbb{T}$ jest $\sigma \in_p \tau$ i jednocześnie dla innych $q \in \mathbb{T}$ jest $\sigma \notin_q \tau$, przynależność σ do τ jest wyznaczona przez dwa zbiory, co można formalnie zapisać tak:

$$\exists_{P \subset \mathbb{T}} \exists_{Q \subset \mathbb{T}} \left(\bigwedge_{p \in P} \sigma \in_p \tau \wedge \bigwedge_{q \in Q} \sigma \notin_q \tau \right). \quad (4.2.1)$$

4.2.2 Równość

Równość pomiędzy metazbiorami definiujemy podobnie jak należenie. Będzie ona odzwierciedlać „przybliżoną równość” opartą na „częściowym należeniu”.

DEFINICJA 4.2.5. Dla $p \in \mathbb{T}$ powiemy, że metazbiór σ jest *równy* meta zbiorowi τ *pod warunkiem* p (ozn. $\sigma \approx_p \tau$), gdy warunek ten wymusza ich równość:

$$\sigma \approx_p \tau \leftrightarrow p \Vdash \sigma = \tau.$$

Jeśli $p = \mathbb{1}$, to piszemy $\sigma \approx \tau$ i mówimy, że metazbiory σ i τ są *równe*.

Inaczej mówiąc $\sigma \approx_p \tau$, gdy w każdej interpretacji C zawierającej p zachodzi $\sigma_C = \tau_C$. Podobnie definiujemy zaprzeczenie równości.

DEFINICJA 4.2.6. Dla $p \in \mathbb{T}$ powiemy, że metazbiory σ, τ są *różne* *pod warunkiem* $p \in \mathbb{T}$ (ozn. $\sigma \not\approx_p \tau$), jeśli zachodzi $p \Vdash \sigma \neq \tau$. Jeżeli $p = \mathbb{1}$, to mówimy, że σ i τ są *różne* i piszemy $\sigma \not\approx \tau$.

Jeśli $\sigma \not\approx_p \tau$, to w każdej interpretacji $C \ni p$ zbiory σ_C i τ_C są różne. Zwróćmy uwagę, że metazbiory, które nie są równe, nie muszą wcale być różne i na odwrót.

FAKT 4.2.3. Jeśli ξ, τ są dowolnymi metazbiorami, to:

$$\begin{aligned} \neg(\xi \approx \tau) &\leftrightarrow \xi \not\approx \tau, \\ \neg(\xi \not\approx \tau) &\leftrightarrow \xi \approx \tau. \end{aligned}$$

Przykład 4.1.2 pokazuje, że podobnie jak w przypadku należenia warunkowego możliwe jest, że

$$\exists_{p,q} (\sigma \approx_p \tau \wedge \sigma \not\approx_q \tau) . \quad (4.2.2)$$

Wiele właściwości klasycznych zbiorów przenosi się na metazbiory, czasem w trochę zmodyfikowanej (uogólnionej?) postaci. Przedstawimy teraz dwa takie proste fakty, które wykorzystamy w dalszej części.

FAKT 4.2.4. Niech σ, μ, τ będą metazbiorami, a $p \in \mathbb{T}$.

$$\begin{aligned} \sigma \epsilon_p \tau \wedge \sigma \approx_p \mu &\rightarrow \mu \epsilon_p \tau , \\ \sigma \epsilon_p \tau \wedge \tau \approx_p \mu &\rightarrow \sigma \epsilon_p \mu . \end{aligned}$$

Dowód. Załóżmy, że $\sigma \epsilon_p \tau \wedge \sigma \approx_p \mu$. Dla dowolnej gałęzi $\mathcal{C} \ni p$ zachodzi $\sigma_{\mathcal{C}} \in \tau_{\mathcal{C}}$ i $\sigma_{\mathcal{C}} = \mu_{\mathcal{C}}$, dlatego również $\mu_{\mathcal{C}} \in \tau_{\mathcal{C}}$. Stąd $\mu \epsilon \tau$.

Analogicznie, jeśli w każdej interpretacji jest $\tau_{\mathcal{C}} = \mu_{\mathcal{C}}$ i $\sigma_{\mathcal{C}} \in \tau_{\mathcal{C}}$, to również $\sigma_{\mathcal{C}} \in \mu_{\mathcal{C}}$, a stąd $\sigma \epsilon \mu$. \square

Ekstensjonalność dla metazbiorów wymaga dodatkowo kwantyfikowania po wszystkich stopniach należenia mniejszych niż stopień równości. Wynika to z faktu, że do równości dwóch metazbiorów (pod warunkiem p) nie wystarcza jedynie, że takie same elementy należą do nich jednocześnie (pod warunkiem p), lecz trzeba jeszcze wziąć pod uwagę wszystkie mniejsze stopnie należenia, a więc takie same elementy muszą do nich jednocześnie należeć w takim samym stopniu (mniejszym niż p).

LEMAT 4.2.5. Niech σ, τ - dowolne metazbiory, $p \in \mathbb{T}$.

$$\sigma \approx_p \tau \leftrightarrow \forall_{\mu} \forall_{q \leq p} (\mu \epsilon_q \sigma \leftrightarrow \mu \epsilon_q \tau) .$$

Dowód. Niech \mathcal{C} będzie gałęzią zawierającą p i niech $x \in \sigma_{\mathcal{C}}$ będzie dowolnym zbiorem z interpretacji $\sigma_{\mathcal{C}}$. Lewa strona równoważności mówi, że wtedy $x \in \tau_{\mathcal{C}}$. I na odwrót: jeśli x jest dowolnym zbiorem z interpretacji $\tau_{\mathcal{C}}$, to $x \in \sigma_{\mathcal{C}}$. Weźmy w takim razie dowolny metazbiór μ i $q \leq p$ takie, że $\mu \epsilon_q \sigma$. Dla każdej gałęzi \mathcal{C}_q przechodzącej przez q mamy $\mu_{\mathcal{C}_q} \in \sigma_{\mathcal{C}_q}$. Ale każda taka gałąź przechodzi również przez p , więc $\mu_{\mathcal{C}_q} \in \tau_{\mathcal{C}_q}$. Stąd $\mu \epsilon_q \tau$, i na odwrót: jeśli $\mu \epsilon_q \tau$, to $\mu \epsilon_q \sigma$.

Założmy teraz prawą stronę i weźmy $x \in \sigma_{\mathcal{C}}$, gdzie x jest jakimś zbiorem, a \mathcal{C} — gałęzią zawierającą p . Fakt 3.3.4 stwierdza istnienie metazbioru $\xi \in \text{dom}(\sigma)$ takiego, że $\xi_{\mathcal{C}} = x$. Istnieje również $r \in \sigma[\xi] \cap \mathcal{C}$ takie, że $\xi \epsilon_r \sigma$. Jeśli $r > p$, to również mamy $\xi \epsilon_p \sigma$ (lemat 4.1.1). Z założenia otrzymujemy $\xi \epsilon_r \tau$, więc $\xi_{\mathcal{C}_r} \in \tau$. Pokazaliśmy więc, że jeśli $x \in \sigma_{\mathcal{C}}$, to $x \in \tau_{\mathcal{C}}$. Podobnie rozumując w drugą stronę otrzymujemy $\sigma_{\mathcal{C}} = \tau_{\mathcal{C}}$. Ponieważ wybraliśmy dowolną gałąź \mathcal{C} zawierającą p więc $\sigma \approx_p \tau$. \square

WNIOSEK 4.2.6. Niech σ, τ - będą dowolnymi metazbiorami.

$$\sigma \approx \tau \leftrightarrow \forall_{\mu} \forall_{q \in \mathbb{T}} (\mu \epsilon_q \sigma \leftrightarrow \mu \epsilon_q \tau) .$$

Symbol $=$ w odniesieniu do metazbiorów oznacza ich równość jako zwykle zbiory, czyli *tożsamość*. Oczywiście, tożsamość implikuje równość w sensie zdefiniowanym wyżej. Z drugiej strony z faktu, że $\tau \neq \eta$ nie wynika $\tau \not\approx \eta$. Co więcej, przykład 4.1.1 dowodzi, że mogą istnieć $\tau \neq \eta$ takie, że $\tau \approx \eta$.

Co to znaczy dla metazbioru być trochę pustym? Inaczej: czym może być ξ takie, że $\xi \approx_p \check{0}$? Jeśli $p = \mathbb{1}$, to żeby dla każdej gałęzi \mathcal{C} było $\xi_{\mathcal{C}} = \emptyset$, musi być $\xi = \emptyset$. Jeśli jednak $p \neq \mathbb{1}$, to wystarczy jeśli dla gałęzi \mathcal{C} zawierającej p będzie $\xi_{\mathcal{C}} = \emptyset$, dla innych gałęzi może być cokolwiek. Zatem żeby $\xi \approx_p \check{0}$, to dla każdego $\eta \in \text{dom}(\xi)$ zbiór $\xi[\eta]$ nie może zawierać elementów porównywalnych z p , a w konsekwencji $\text{ran}(\xi)$ może składać się tylko z warunków nieporównywalnych z p . Na przykład dla $p = [0]$, $q = [1]$ i $\xi = \{ \langle \check{1}, q \rangle \}$ jest $\xi \approx_p \emptyset$, bo $\text{ran}(\xi) = \{q\}$ a $p \perp q$. Tak więc ξ jest p -pusty.

4.2.3 Zawieranie

Zawieranie również definiujemy poprzez odwołanie się do wymuszania, czyli pośrednio do interpretacji. Tak jak należenie i równość, zawieranie się metazbiorów będzie uwzględniać „częściową przynależność”.

DEFINICJA 4.2.7. Dla $p \in \mathbb{T}$ powiemy, że metazbiór τ zawiera meta zbiór σ pod warunkiem p , gdy warunek ten wymusza $\sigma \subset \tau$:

$$\sigma \subseteq_p \tau \quad \leftrightarrow \quad p \Vdash \sigma \subset \tau .$$

Podobnie jak przy należeniu i równości, jeśli $p = \mathbb{1}$, to piszemy $\sigma \subseteq \tau$ i mówimy, że τ zawiera σ lub σ jest *metapodzbiorem* τ .

Jeśli $\sigma \subset \tau$, to oczywiście $\sigma \subseteq \tau$. Implikacja odwrotna nie zachodzi, co pokazuje przykład.

PRZYKŁAD 4.2.1: Niech $\tau = \{ \check{0}^{\mathbb{1}}, \check{1}^{\mathbb{1}} \}$ i $\sigma = \{ \check{0}^{\mathbb{1}}, \check{1}^p \}$ dla jakiegoś $p \neq \mathbb{1}$. Mamy $\tau_{\mathcal{C}} = \{0, 1\}$ oraz

$$\sigma_{\mathcal{C}} = \begin{cases} \{0\} & p \notin \mathcal{C} , \\ \{0, 1\} & p \in \mathcal{C} . \end{cases}$$

W każdym przypadku $\sigma_{\mathcal{C}} \subset \tau_{\mathcal{C}}$, więc $\sigma \subseteq \tau$. Ale $\sigma \not\subset \tau$, bo $\langle \check{1}, p \rangle \in \sigma$ i $\langle \check{1}, p \rangle \notin \tau$.

Podobnie — w ogólnym przypadku — nie jest prawdą, że

$$\xi \subseteq \eta \rightarrow \text{dom}(\xi) \subset \text{dom}(\eta) . \quad (4.2.3)$$

PRZYKŁAD 4.2.2: Weźmy $\eta = \{ \check{1}^{\mathbb{1}} \}$ i $\xi = \{ \check{1}^{\mathbb{1}}, \zeta^{\mathbb{1}} \}$, gdzie $\zeta = \{ \check{0}^p \}$ dla pewnego $p \neq \mathbb{1}$. Mamy: $\text{dom}(\eta) = \{ \check{1} \}$, oraz $\text{dom}(\xi) = \{ \check{1}, \zeta \}$, więc oczywiście $\text{dom}(\xi) \not\subset \text{dom}(\eta)$. Jednocześnie $\eta_{\mathcal{C}} = \{1\}$ i $\xi_{\mathcal{C}} = \{1, \zeta_{\mathcal{C}}\} = \{1\}$, bo

$$\zeta_{\mathcal{C}} = \begin{cases} 1 = \{ \emptyset \} & \text{gdy } p \in \mathcal{C} , \\ \emptyset & \text{gdy } p \notin \mathcal{C} . \end{cases}$$

Więc dalej mamy $\xi \subseteq \eta$, choć dziedziny są różne.

W rozdziale 5 będziemy rozważać klasę metazbiorów, na której powyższa implikacja jest spełniona (patrz lemat 5.2.4).

Przedstawimy teraz dwie proste, typowe właściwości klasycznych zbiorów, które mają odpowiedniki dla metazbiorów. Będziemy z nich korzystać w dalszej części.

LEMAT 4.2.7. Niech σ, τ - dowolne metazbiory, $p \in \mathbb{T}$.

$$\sigma \approx_p \tau \quad \leftrightarrow \quad \sigma \subseteq_p \tau \wedge \tau \subseteq_p \sigma .$$

Dowód. Przez przejście do interpretacji i odwołanie się do analogicznego związku dla zwykłych zbiorów. \square

WNIOSEK 4.2.8. Dla $\sigma, \tau \in \mathfrak{M}$ jest

$$\sigma \approx \tau \quad \leftrightarrow \quad \sigma \subseteq \tau \wedge \tau \subseteq \sigma .$$

Dowód. Podstawić $p = \mathbb{1}$ w powyższym lemacie. \square

Relacja zawierania dla metazbiorów jest przechodnia.

LEMAT 4.2.9. Dla $\tau, \sigma, \eta \in \mathfrak{M}$ i $p \in \mathbb{T}$, jeśli $\tau \subseteq_p \sigma$ i $\sigma \subseteq_p \eta$, to $\tau \subseteq_p \eta$.

Dowód. Niech \mathcal{C} będzie gałęzią zawierającą p . Z założenia mamy $\tau_{\mathcal{C}} \subseteq \sigma_{\mathcal{C}}$ oraz $\sigma_{\mathcal{C}} \subseteq \eta_{\mathcal{C}}$, zatem również $\tau_{\mathcal{C}} \subseteq \eta_{\mathcal{C}}$. Skoro $\tau_{\mathcal{C}} \subseteq \eta_{\mathcal{C}}$ dla każdej gałęzi $\mathcal{C} \ni p$, to $\tau \subseteq_p \eta$. \square

Rozdział 5

Metazbiory pierwszego poziomu

Bardzo ważną klasę metazbiorów stanowią metazbiory pierwszego poziomu, które zbadamy w tym rozdziale. Można je postrzegać jako metazbiory, których elementami są zwykłe zbiory. Rozmycie (relacji należenia) zachodzi tu tylko na pierwszym poziomie hierarchii należenia.

Dla metazbiorów pierwszego poziomu zdefiniujemy operacje algebraiczne i pokażemy, że spełniają one aksjomaty algebry Boole'a. Dla dziedzicznie skończonych metazbiorów pierwszego poziomu wprowadzimy inne, prostsze obliczeniowo, charakteryzacje podstawowych relacji.¹ Będą one miały fundamentalne znaczenie dla komputerowych implementacji metazbiorów.

Skąd się wzięła potrzeba ograniczenia pewnych rozważań do metazbiorów szczególnej postaci? Rozważmy poniższy przykład, który ukazuje sedno problemu.

PRZYKŁAD 5.0.3: Elementy metazbiorów mogą się zmieniać w zależności od interpretacji. Rozważmy dla przykładu $\tau = \{\langle \emptyset, p \rangle\}$ oraz $\sigma = \{\langle \tau, \mathbb{1} \rangle\}$, gdzie p jest dowolnym warunkiem różnym od $\mathbb{1}$. W zależności od wybranej gałęzi \mathcal{C} , zbiór $\sigma_{\mathcal{C}}$ może mieć różną zawartość. Zawsze będzie on jednoelementowy, jednakże ten element będzie się zmieniał.

$$\sigma_{\mathcal{C}} = \{\tau_{\mathcal{C}}\} = \begin{cases} \{\{\emptyset\}\} & \text{gdy } p \in \mathcal{C}, \quad \text{bo } \tau_{\mathcal{C}} = \{\emptyset\} , \\ \{\emptyset\} & \text{gdy } p \notin \mathcal{C}, \quad \text{bo } \tau_{\mathcal{C}} = \emptyset . \end{cases} \quad (5.0.1)$$

Ta zmienność utrudnia analizę metazbiorów i ogranicza możliwości zastosowań, bowiem często chcielibyśmy mieć metazbiory, których elementy będą identyczne w każdej interpretacji. Cechę tę mają właśnie metazbiory pierwszego poziomu.

¹ W świetle najnowszych, niepublikowanych badań założenie o dziedzicznej skończoności można opuścić we wszystkich lematach i twierdzeniach tej części. W szczególności nie jest ono konieczne do wprowadzenia nowych charakteryzacji dla relacji.

5.1 Podstawowe definicje

Definicje podstawowych relacji i operacji dla metazbiorów pierwszego poziomu są wyrażone w terminach podzbiorów drzewa binarnego. Dlatego w tej sekcji, oprócz definicji metazbiorów pierwszego i wyższych poziomów, wprowadzimy kilka pojęć niezbędnych do określenia relacji i operacji. Zbadamy też ich podstawowe właściwości.

5.1.1 Poziomy metazbiorów

Elementami metazbiorów pierwszego poziomu są pary postaci $\langle \check{\sigma}, p \rangle$. Pierwszym elementem takiej pary jest metazbiór kanoniczny, dzięki czemu elementy interpretacji takich metazbiorów będą zawsze takie same, niezależnie od gałęzi definiującej interpretację, gdyż metazbiór kanoniczny jest w każdej interpretacji tym samym zbiorem (por. fakt 3.3.1).

DEFINICJA 5.1.1. Metazbiorem *pierwszego poziomu* nazywamy meta zbiór postaci \emptyset lub meta zbiór postaci:

$$\tau = \{ \langle \check{\sigma}, p \rangle : p \in \mathbb{T} \wedge \check{\sigma} \in \mathfrak{M}^c \} .$$

Klasę metazbiorów pierwszego poziomu oznaczamy symbolem \mathfrak{M}^1 . Ważniejsza — z punktu widzenia zastosowań — jest jej podklasa złożona z metazbiorów dziedzicznie skończonych. Oznaczamy ją symbolem $\mathfrak{M}\mathfrak{F}^1$. Zatem:

$$\mathfrak{M}\mathfrak{F}^1 = \mathfrak{M}\mathfrak{F} \cap \mathfrak{M}^1 . \quad (5.1.1)$$

Elementami rozważanych tu metazbiorów klasy $\mathfrak{M}\mathfrak{F}^1$ będą meta zbiory kanoniczne, które są dziedzicznie skończone. Przyjmujemy oznaczenie $\mathfrak{M}\mathfrak{F}^c$ na tego typu metazbiory. A zatem:

$$\mathfrak{M}\mathfrak{F}^c = \mathfrak{M}\mathfrak{F} \cap \mathfrak{M}^c . \quad (5.1.2)$$

Analogicznie do metazbiorów pierwszego poziomu możemy zdefiniować metazbiory wyższych poziomów.

DEFINICJA 5.1.2. Niech $n \in \mathbb{N}$ i $n > 1$. Metazbiór pusty lub postaci:

$$\tau^n = \{ \langle \check{\sigma}^{n-1}, p \rangle : p \in \mathbb{T} \wedge \check{\sigma}^{n-1} \in \mathfrak{M}^{n-1} \} .$$

nazywamy metazbiorem *n-tego poziomu*.

Klasę metazbiorów *n-tego poziomu* oznaczamy symbolem \mathfrak{M}^n . Analogicznie jak w przypadku metazbiorów pierwszego poziomu możemy wyróżnić ich podklasy składające się z metazbiorów dziedzicznie skończonych: przez $\mathfrak{M}\mathfrak{F}^n$ oznaczamy klasę metazbiorów dziedzicznie skończonych *n-tego poziomu*.

$$\mathfrak{M}\mathfrak{F}^n = \mathfrak{M}\mathfrak{F} \cap \mathfrak{M}^n . \quad (5.1.3)$$

5.1.2 Pokrywanie podzbiorów \mathbb{T}

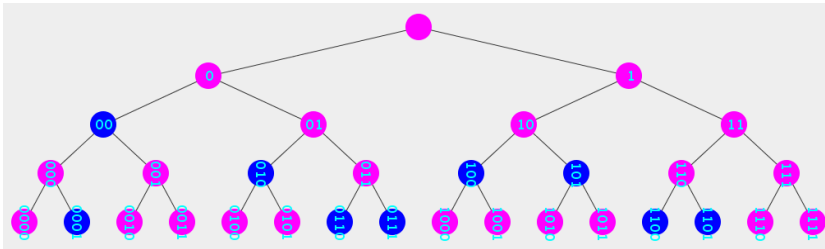
Lemat 4.1.1 mówi, że jeśli pewna właściwość zachodzi dla warunku p , to również zachodzi dla wszystkich silniejszych. Z kolei lemat 4.1.2 pokazuje rzecz

odwrotną: jeśli pewna właściwość jest spełniona dla zbioru warunków, który zawiera skończony antyłańcuch maksymalny poniżej p , to ta właściwość przenosi się na p . Zatem można powiedzieć, że pojedynczy warunek i pewien szczególny zbiór warunków są w jakiś sposób sobie równoważne.

Poniżej sformalizujemy tę relację pomiędzy podzbiorem drzewa \mathbb{T} i jego elementem w taki sposób, by można ją było potem łatwo badać i zaimplementować.

DEFINICJA 5.1.3. Niech $R \subset \mathbb{T}$ i $p \in \mathbb{T}$. Powiemy, że R rozpina p (ozn. $R \vdash p$), gdy R zawiera skończony antyłańcuch maksymalny poniżej p . Jeśli dodatkowo $p = \mathbb{1}$, to mówimy, że R rozpina \mathbb{T} (ozn. $R \vdash$).

Zauważmy, że jeśli R rozpina p , to każdy element drzewa \mathbb{T} porównywalny z p jest porównywalny z jakimś elementem ze zbioru R .



Rysunek 5.1: Powyższy zbiór rozpina $[0]$, ale nie rozpina $[1]$

Na rys. 5.1 widzimy zbiór, który rozpina $[0]$, $[10]$ i $[110]$, ale nie rozpina $[1]$. Zauważmy, że nie jest on antyłańcuchem poniżej $[0]$, ale zawiera jeden, który jest maksymalny.

Zgodnie z definicją 5.1.3 zbiór $R \subset \mathbb{T}$ rozpina warunek $p \in \mathbb{T}$, gdy R zawiera skończony antyłańcuch maksymalny, którego wszystkie elementy leżą poniżej p , jednak nie wszystkie elementy z R muszą być poniżej p . Oznacza to sytuację, w której każda gałąź w drzewie \mathbb{T} przechodząca przez p zawiera również jakiś element ze zbioru R . Zauważmy, że ta cenna właściwość jest również spełniona w przypadku, gdy zbiór R zawiera warunek słabszy od p , a nie zawiera antyłańcucha, ani być może w ogóle żadnych elementów, poniżej p . Poniższa definicja łączy te dwie cechy.

DEFINICJA 5.1.4. Niech $R \subset \mathbb{T}$ i $p \in \mathbb{T}$. Powiemy, że R pokrywa p (ozn. $R \mid p$), gdy R rozpina p lub R zawiera warunek słabszy od p . A zatem:

$$R \mid p \leftrightarrow R \vdash p \vee \exists_{q \in R} q \geq p .$$

Jeśli $R = \emptyset$, to zdanie $R \mid p$ (czyli $\emptyset \mid p$) jest fałszywe dla każdego $p \in \mathbb{T}$. Zauważmy, że $\{p\}$ rozpina p , jak również zawiera warunek słabszy od p , a więc $\{p\}$ pokrywa p . Pokrywanie jest propagowane w dół, o czym mówi następujący lemat.

LEMAT 5.1.1. Niech $R \subset \mathbb{T}$ będzie dowolnym podzbiorem drzewa \mathbb{T} , a p i $q \in \mathbb{T}$ niech będą warunkami takimi, że $q \leq p$. Jeśli $R \mid p$, to $R \mid q$.

Dowód. Jeśli $R = \emptyset$, to $R \mid p$ jak i $R \mid q$ są fałszywe. W przeciwnym przypadku, jeśli R zawiera warunek słabszy od p , to również zawiera warunek słabszy od q (bo $q \leq p$) i wtedy $R \mid q$. W przeciwnym przypadku R zawiera skończony

antyłańcuch maksymalny A poniżej p . Jeśli A zawiera warunek $r \geq q$, to $R \mid q$. Jeśli nie, to $A_q = \{r \in A : r \leq q\}$ musi tworzyć antyłańcuch maksymalny (każdy warunek poniżej q musi być porównywalny z jakimś warunkiem z A , a ten nie zawierał nic powyżej q). Oczywiście $A_q \subset A$ jest skończony i stąd $R \mid q$. \square

Pokrywanie może być również propagowane w górę, jeśli pokrywane warunki tworzą skończony antyłańcuch maksymalny poniżej innego warunku.

LEMAT 5.1.2. Niech $p \in \mathbb{T}$, a $A \subset \mathbb{T}$ będzie skończonym antyłańcuchem maksymalnym poniżej p . Jeśli $R \subset \mathbb{T}$ jest zbiorem warunków takim, że $\forall_{q \in A} R \mid q$, to również $R \mid p$.

Dowód. Jeśli R zawiera warunek $r \geq p$, to oczywiście $R \mid p$. Załóżmy więc, że $\forall_{r \in R} r \top p \rightarrow r < p$. Niech dla każdego $q \in A$ zbiór $R_q \subset R$ świadczy, że $R \mid q$, tzn. albo R_q składa się z warunku słabszego od q , albo R_q jest skończonym antyłańcuchem maksymalnym poniżej q zawartym w R . Wtedy $R_p = \bigcup_{q \in A} R_q$ zawiera skończony antyłańcuch maksymalny poniżej p . Jest nim zbiór R_p^{\max} złożony z elementów maksymalnych w R_p . Jest on skończony, bo A jest skończony i każde R_q też. Jest on antyłańcuchem, bo nie zawiera elementów parami porównywalnych. Żeby zauważyć, że R_p^{\max} jest antyłańcuchem maksymalnym poniżej p , weźmy jakiś warunek $r \leq p$. Pokażemy, że r jest porównywalny z jakimś elementem zbioru R_p^{\max} . Istnieje $q \in A$ porównywalne z r , gdyż A jest antyłańcuchem maksymalnym poniżej p . Istnieje również $r_q \in R_q$ porównywalne z r , gdyż R_q zawiera elementy leżące na wszystkich gałęziach przechodzących przez q , w szczególności r_q leży na tej, która przechodzi przez r . Oczywiście $r_q \in R_p$, więc w zbiorze R_p^{\max} istnieje element $r_q^{\max} \geq r_q$. Jest on porównywalny z r gdyż r_q^{\max}, r_q, q i r leżą na jednej gałęzi. Zatem $R_p \mid p$, a co za tym idzie $R \mid p$. \square

5.1.3 Równoważność podzbiorów \mathbb{T}

Zbiory, które wzajemnie pokrywają swoje elementy mają szczególnie cenne właściwości, które teraz zbadamy. Takie zbiory będą grały kluczową rolę przy określaniu niektórych relacji i operacji.

DEFINICJA 5.1.5. Niech Q, R będą dowolnymi podzbiórmi drzewa \mathbb{T} . Powiemy że Q i R są *równoważne*, jeśli

$$\forall_{q \in Q} R \mid q \wedge \forall_{r \in R} Q \mid r .$$

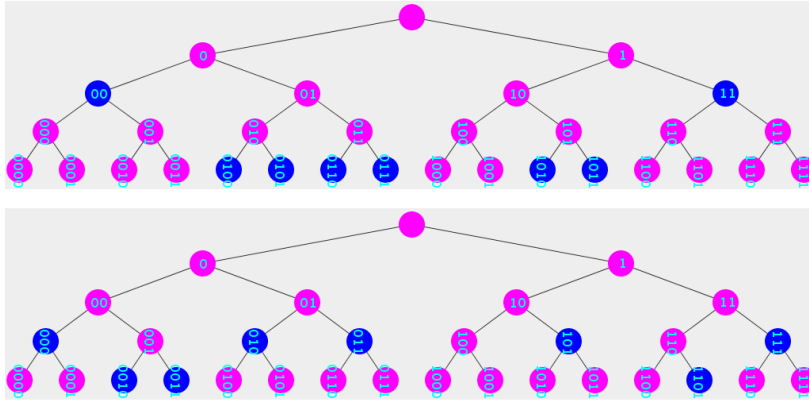
Równoważność zbiorów Q i R oznaczamy symbolem $Q \parallel R$.

Zauważmy, że zdania $Q \parallel \emptyset$ i $\emptyset \parallel R$ są zawsze fałszywe dla niepustych Q, R (bo $\emptyset \mid p$ jest fałszywe), oraz że zdanie $\emptyset \parallel \emptyset$ jest prawdziwe.

Rysunek 5.2 przedstawia dwa zbiory równoważne poniżej [0]. Nie są one równoważne poniżej [1] ani [11], ale dodatkowo są równoważne poniżej [10].

W praktyce równoważność zbiorów Q i R oznacza tyle, że jeśli gałąź C w \mathbb{T} zawiera jakiś warunek z Q , to musi również zawierać jakiś warunek z R , i na odwrót.

Dla danego zbioru warunków $W \subset \mathbb{T}$ często będziemy zainteresowani tymi jego elementami, które są porównywalne z jakimś warunkiem p .

Rysunek 5.2: Zbiory równoważne poniżej $[0]$, ale nie równoważne poniżej $[1]$

DEFINICJA 5.1.6. Dla $W \subset \mathbb{T}$ i $p \in \mathbb{T}$ *obcięciem* zbioru W do warunku p nazywamy jego podzbiór złożony z elementów porównywalnych z p . Obcięcie W do p oznaczamy symbolem $W \upharpoonright_p$. Zatem:

$$W \upharpoonright_p = \{w \in W : w \top p\} .$$

Zauważmy, że $W \upharpoonright_{\mathbf{1}} = W$. Rozważmy jakąś gałąź $\mathcal{C} \subset \mathbb{T}$ przechodzącą przez warunek p . Jeśli zawiera ona elementy ze zbioru $W \subset \mathbb{T}$, to są to elementy jego obcięcia do p . Innymi słowy:

$$\mathcal{C} \cap W \subset W \upharpoonright_p . \quad (5.1.4)$$

To wyjaśnia również dlaczego w definicji obcięcia nie ograniczamy się tylko do warunków silniejszych od p , ale bierzemy wszystkie porównywalne z p , a więc również leżące powyżej.

Równoważność podzbiorów drzewa \mathbb{T} można ograniczyć do warunków porównywalnych z danym warunkiem $p \in \mathbb{T}$. Takie obcięcie będzie potrzebne do wysłowienia warunkowych relacji dla metazbiorów w terminach podzbiorów \mathbb{T} .

DEFINICJA 5.1.7. Niech $Q, R \subset \mathbb{T}$ będą dowolnymi podzbiarami, a $p \in \mathbb{T}$. Jeśli $Q \upharpoonright_p \neq \emptyset \neq R \upharpoonright_p$, to powiemy że Q i R są *równoważne poniżej p* , jeżeli:

$$\forall_{q \in Q \upharpoonright_p} R \upharpoonright_p \mid \min(p, q) \wedge \forall_{r \in R \upharpoonright_p} Q \upharpoonright_p \mid \min(p, r) .$$

Jeżeli $Q \upharpoonright_p = \emptyset = R \upharpoonright_p$, to przyjmujemy, że Q i R są *równoważne poniżej p* . Jeśli natomiast $Q \upharpoonright_p = \emptyset \neq R \upharpoonright_p$ lub $Q \upharpoonright_p \neq \emptyset = R \upharpoonright_p$, to przyjmujemy, że Q i R *nie są równoważne poniżej p* . Równoważność zbiorów Q i R poniżej warunku p oznaczamy symbolem $Q \parallel_p R$.

Zauważmy, że dla dowolnych $Q, R \subset \mathbb{T}$ zdania $Q \parallel R$ oraz $Q \parallel_{\mathbf{1}} R$ są równoważne. Dla zbiorów $Q, R \subset \mathbb{T}$, które są równoważne poniżej p , jeśli Q zawiera warunek $q \geq p$, to — żeby było spełnione $R \upharpoonright_p \mid \min(p, q)$ — zbiór R musi zawierać warunek $r \geq p$ lub skończony antyłańcuch maksymalny poniżej p . Z drugiej strony, jeśli $q \in Q$ jest takie, że $q \leq p$, to $R \upharpoonright_p$ pokrywa q albo przy pomocy warunku powyżej q , albo przy pomocy antyłańcucha poniżej q .

5.2 Podstawowe relacje

Definicje podstawowych relacji mnogościowych pomiędzy metazbiorami odwołują się do ich interpretacji i do relacji wymuszania (sekcja 4.2). Komputerowa analiza w ten sposób zdefiniowanych relacji byłaby uciążliwa, albowiem wymagałaby sprawdzenia dużej (potencjalnie nieskończonej) liczby interpretacji. W tym podrozdziale wprowadzimy inne charakteryzacje tych relacji dla dziedzicznie skończonych metazbiorów pierwszego poziomu.² Będą one odwoływać się do struktury metazbiorów będących w relacji, a dokładniej do związków pomiędzy obrazami elementów potencjalnych, które są zbiorami elementów drzewa. Dzięki temu będzie można zaprojektować proste i efektywne algorytmy operujące na takich metazbiorach.

5.2.1 Należenie

Elementami potencjalnymi metazbiorów pierwszego poziomu są meta zbiory kanoniczne, a więc relacja należenia będzie zachodzić pomiędzy tego rodzaju zbiorami. Metazbiór kanoniczny jest w każdej interpretacji taki sam (fakt 3.3.3), więc nie będziemy musieli brać pod uwagę możliwych różnych interpretacji tak, jak ma to miejsce w przypadku zwykłych metazbiorów.

Poniższy lemat pokazuje ważną cechę metazbiorów pierwszego poziomu: metazbiór kanoniczny, który należy warunkowo do metazbioru pierwszego poziomu jest również elementem jego dziedziny.

LEMAT 5.2.1. Niech $\check{\sigma} \in \mathfrak{M}^c$ i $\tau \in \mathfrak{M}^1$. Jeśli istnieje $p \in \mathbb{T}$ takie, że $\check{\sigma} \epsilon_p \tau$, to $\check{\sigma} \in \text{dom}(\tau)$.

Dowód. Jeśli $\check{\sigma} \epsilon_p \tau$, to (def. 4.2.1) dla każdej gałęzi $\mathcal{C} \ni p$ jest $\check{\sigma}_{\mathcal{C}} \in \tau_{\mathcal{C}}$. Niech $x = \check{\sigma}_{\mathcal{C}}$ będzie tym jedynym — ze względu na fakt 3.3.3 — zbiorem, który jest równy wszystkim interpretacjom $\check{\sigma}$ zadanym przez gałęzie zawierające p . Na mocy faktu 3.3.5 istnieje dokładnie jeden metazbiór kanoniczny $\check{x} \in \text{dom}(\tau)$ taki, że $\check{x}_{\mathcal{C}} = x$ i musi to być $\check{\sigma}$. Stąd $\check{\sigma} \in \text{dom}(\tau)$. \square

W ogólnym przypadku przedstawiony wyżej związek nie zachodzi.

PRZYKŁAD 5.2.1: Jeśli weźmiemy $\tau = \{\langle \check{0}, \mathbb{1} \rangle, \langle \check{1}, \mathbb{1} \rangle\}$, i $\sigma = \{\langle \check{0}, [0] \rangle\}$, to $\sigma \in \tau$, ale $\sigma \notin \text{dom}(\tau)$. Rzeczywiście, jeśli \mathcal{C} jest gałęzią zawierającą $[0]$, to $\sigma_{\mathcal{C}} = \{0\} = 1$ oraz $\tau_{\mathcal{C}} = \{0, 1\}$. Jeśli natomiast $\mathcal{C} \ni [1]$, to $\sigma_{\mathcal{C}} = \emptyset = 0$ oraz $\tau_{\mathcal{C}} = \{0, 1\}$. W obu przypadkach $\sigma_{\mathcal{C}} \in \tau_{\mathcal{C}}$, więc $\sigma \in \tau$. Jednak gołym okiem widać, że $\sigma \notin \text{dom}(\tau)$.

Teraz podamy charakteryzację należenia warunkowego dla dziedzicznie skończonych metazbiorów pierwszego poziomu.

TWIERDZENIE 5.2.2. Jeśli $\tau \in \mathfrak{M}\mathfrak{F}^1$, $\check{\sigma} \in \mathfrak{M}\mathfrak{F}^c$, i $p \in \mathbb{T}$, to $\check{\sigma} \epsilon_p \tau$ wtedy i tylko wtedy, gdy $\tau[\check{\sigma}]$ pokrywa p :

$$\check{\sigma} \epsilon_p \tau \iff \tau[\check{\sigma}] \mid p .$$

Dowód. Załóżmy lewą stronę. Lemat 5.2.1 mówi, że $\check{\sigma} \in \text{dom}(\tau)$, a zatem $\tau[\check{\sigma}]$ jest niepuste. Jeśli $\check{\sigma} \epsilon_p \tau$, to (def. 4.2.1) dla każdej gałęzi \mathcal{C} zawierającej p jest $\check{\sigma}_{\mathcal{C}} \in \tau_{\mathcal{C}}$. W takim razie, dla każdej gałęzi $\mathcal{C} \ni p$ musi istnieć $p_{\mathcal{C}} \in \mathcal{C}$ takie, że

²Właściwie dla metazbiorów pierwszego poziomu, por. przypis na str. 29.

$\langle \check{\sigma}, p_C \rangle \in \tau$ (por. fakty 3.3.3 i 3.3.5). Zbiór tych p_C jest obcięciem $\tau[\check{\sigma}] \upharpoonright_p$ zawartym w $\tau[\check{\sigma}]$. Żeby pokazać pokrywanie trzeba w $\tau[\check{\sigma}]$ (a dokładniej w $\tau[\check{\sigma}] \upharpoonright_p$) wskazać warunek słabszy od p lub skończony antylańcuch maksymalny poniżej p . Załóżmy, że obraz $\tau[\check{\sigma}]$ nie zawiera warunku słabszego od p . Wtedy zawiera on co najmniej jeden element z każdej gałęzi C przechodzącej przez p , który jest silniejszy od p . Zatem $\tau[\check{\sigma}] \upharpoonright_p = \{q: q < p \wedge \langle \check{\sigma}, q \rangle \in \tau\}$ zawiera antylańcuch maksymalny A , którym jest zbiór jego elementów maksymalnych $\max(\tau[\check{\sigma}] \upharpoonright_p)$. A jest maksymalny, bo zawiera element z każdej gałęzi $C \ni p$. Jest on skończony, bo dziedziczna skończoność τ implikuje skończoność $\tau[\check{\sigma}] \supset A$. Pokazaliśmy, że $\tau[\check{\sigma}]$ zawiera skończony antylańcuch maksymalny poniżej p (o ile nie zawiera warunku słabszego od p), a więc pokrywa p .

Założmy teraz prawą stronę. Jeśli $\tau[\check{\sigma}]$ zawiera $q \geq p$, to dla każdej gałęzi C przechodzącej przez q mamy $\check{\sigma}_C \in \tau_C$, czyli $q \Vdash \check{\sigma} \in \tau$. Lemat 4.1.1 mówi, że w tej sytuacji również $p \Vdash \check{\sigma} \in \tau$, czyli $\check{\sigma} \in_p \tau$. Jeśli $\tau[\check{\sigma}]$ zawiera skończony antylańcuch maksymalny A poniżej p , to stosujemy lemat 4.1.2. Ponieważ dla każdego $q \in A$ jest $q \Vdash \check{\sigma} \in \tau$ (tak jest dla każdego elementu $\tau[\check{\sigma}]$), to również $p \Vdash \check{\sigma} \in \tau$, czyli $\check{\sigma} \in_p \tau$. Stąd $\check{\sigma} \in_p \tau$. \square

W skrócie: $\check{\sigma} \in_p \tau$, gdy każda gałąź przechodząca przez p zawiera warunek z obrazu $\tau[\check{\sigma}]$, co gwarantuje pojawienie się $\check{\sigma}_C$ w interpretacji τ_C .

Jeśli $p = \mathbb{1}$, to powyższe twierdzenie przyjmuje następującą postać.

WNIOSEK 5.2.3. Jeśli $\tau \in \mathfrak{M}\mathfrak{F}^1$, a $\check{\sigma} \in \mathfrak{M}\mathfrak{F}^c$, to $\check{\sigma} \in \tau$ wtedy i tylko wtedy, gdy $\tau[\check{\sigma}]$ rozpina \mathbb{T} :

$$\check{\sigma} \in \tau \quad \leftrightarrow \quad \tau[\check{\sigma}] \vdash .$$

Dowód. Przyjmując $p = \mathbb{1}$ w tw. 5.2.2 otrzymujemy $\check{\sigma} \in \tau \leftrightarrow \tau[\check{\sigma}] \mid \mathbb{1}$. Ale dla dowolnego $S \subset \mathbb{T}$ oczywiście $S \mid \mathbb{1}$ (S pokrywa $\mathbb{1}$) jest równoważne $S \vdash \mathbb{1}$ (S rozpina $\mathbb{1}$), czyli $S \vdash (S \text{ rozpina } \mathbb{T})$. \square

Jak widać, aby rozstrzygnąć należenie $\check{\sigma}$ do τ wystarczy sprawdzić, czy $\tau[\check{\sigma}]$ zawiera antylańcuch maksymalny (będzie on skończony, jeśli τ jest dziedzicznie skończony). Nie jest to problem złożony obliczeniowo, zwłaszcza jeśli weźmiemy pod uwagę, że warunki można traktować jako ciągi binarne reprezentujące liczby naturalne bądź dwójkowo-wymierne. W przypadku należenia pod warunkiem p należy zrobić to samo dla poddrzewa, którego korzeniem jest p .

5.2.2 Zawieranie

W ogólnym przypadku z zawierania się metazbiorów nie wynika zawieranie się ich dziedzin, tzn. nie jest prawdą, że $\xi \in \eta \rightarrow \text{dom}(\xi) \subset \text{dom}(\eta)$ (por. przykład 4.2.2). Jednak dla metazbiorów pierwszego poziomu tak jest, co teraz pokażemy.

LEMAT 5.2.4. Niech $\xi, \eta \in \mathfrak{M}^1$. Jeśli $\xi \in \eta$, to $\text{dom}(\xi) \subset \text{dom}(\eta)$.

Dowód. Przypuśćmy przeciwnie: niech $\check{\mu} \in \text{dom}(\xi)$ i $\check{\mu} \notin \text{dom}(\eta)$. Jeśli $p \in \xi[\check{\mu}]$, a $C \ni p$ jest gałęzią, to $\check{\mu}_C \in \xi_C$. Gdyby $\check{\mu}_C \in \eta_C$, to na mocy faktu 3.3.5 byłoby $\check{\mu} \in \text{dom}(\eta)$. Z kolei $\check{\mu}_C \notin \eta_C$ przeczy założeniu, że $\xi \in \eta$, czyli $\xi_C \subset \eta_C$. \square

Implikacja odwrotna oczywiście nie zachodzi — nawet metazbiory pierwszego poziomu o tych samych dziedzinach nie muszą być w żadnej relacji zawierania.

PRZYKŁAD 5.2.2: Weźmy dla przykładu

$$\xi = \{ \langle \check{\sigma}, [0] \rangle, \langle \check{\tau}, [1] \rangle \} \quad \text{oraz} \quad \eta = \{ \langle \check{\sigma}, [1] \rangle, \langle \check{\tau}, [0] \rangle \} ,$$

przy czym niech $\check{\sigma} = \check{0} = \emptyset$, a $\check{\tau} = \check{1} = \{ \check{0}^1 \}$. Oczywiście, dziedziny są równe, bo $\text{dom}(\xi) = \text{dom}(\eta) = \{ \check{\sigma}, \check{\tau} \}$. Jednak zawieranie warunkowe nie zachodzi dla żadnego warunku. Interpretacje dzielą się w tym przykładzie na dwie kategorie, wyznaczone przez gałęzie przechodzące przez $p = [0]$ lub $q = [1]$. Wszystkie gałęzie \mathcal{C}_p przechodzące przez p dają te same interpretacje: $\xi_{\mathcal{C}_p} = \{ \check{\sigma}_{\mathcal{C}_p} \}$ oraz $\eta_{\mathcal{C}_p} = \{ \check{\tau}_{\mathcal{C}_p} \}$. Podobnie, gałęzie \mathcal{C}_q przechodzące przez q dają takie interpretacje: $\xi_{\mathcal{C}_q} = \{ \check{\tau}_{\mathcal{C}_q} \}$ oraz $\eta_{\mathcal{C}_q} = \{ \check{\sigma}_{\mathcal{C}_q} \}$. Widać, że w żadnym przypadku nie zachodzi tu zawieranie warunkowe.

Nie jest też prawdą, że dla $\xi, \eta \in \mathfrak{M}^1$ zachodzi $\xi \subseteq \eta \rightarrow \xi \subset \eta$. Co prawda, dziedziny się zawierają, ale warunki z jakimi elementy potencjalne występują w parach mogą być różne.

PRZYKŁAD 5.2.3: Niech dla przykładu $\check{\sigma}$ będzie dowolnym metazbiorem kanonicznym i $\xi = \{ \langle \check{\sigma}, \mathbb{1} \rangle \}$, a $\eta = \{ \langle \check{\sigma}, [0] \rangle, \langle \check{\sigma}, [1] \rangle \}$. Oczywiście $\xi \subseteq \eta$, bo $\xi_{\mathcal{C}} \subset \eta_{\mathcal{C}}$ dla dowolnej gałęzi \mathcal{C} , jednak $\xi \not\subseteq \eta$ (por. przykł. 4.2.1).

Jeśli $\xi, \eta \in \mathfrak{M}^1$ i $\xi \subseteq_p \eta$, to nie zachodzi żaden związek pomiędzy dziedzinami tych metazbiorów poza tym, że muszą mieć część wspólną.

PRZYKŁAD 5.2.4: Dla przykładu rozważmy $\tau = \{ \check{\sigma}^p, \check{\mu}^q \}$ i $\xi = \{ \check{\sigma}^p, \check{\eta}^r \}$, gdzie p, q, r są nieporównywalne (np.: $p = [00]$, $q = [10]$, $r = [11]$). Mamy $\tau \approx_p \xi$, więc $\tau \subseteq_p \xi$, jak również $\xi \subseteq_p \tau$, bo dla $\mathcal{C} \ni p$ jest $\tau_{\mathcal{C}} = \{ \check{\mu}_{\mathcal{C}} \} = \xi_{\mathcal{C}}$. Jednak $\text{dom}(\tau) = \{ \check{\sigma}, \check{\mu} \}$, a $\text{dom}(\xi) = \{ \check{\sigma}, \check{\eta} \}$.

Przedstawimy teraz charakteryzację zawierania warunkowego dla dziedzicznie skończonych metazbiorów pierwszego poziomu w postaci zależności pomiędzy obrazami elementów potencjalnych.

TWIERDZENIE 5.2.5. Niech $\tau, \sigma \in \mathfrak{M}\mathfrak{F}^1$, a $p \in \mathbb{T}$. Wtedy:

$$\tau \subseteq_p \sigma \quad \leftrightarrow \quad \forall_{\check{\eta} \in \text{dom}(\tau)} \forall_{q \in \tau[\check{\eta}] \upharpoonright_p} \sigma[\check{\eta}] \mid \min(p, q) .$$

Zatem τ jest zawarte w σ pod warunkiem p wtedy i tylko wtedy, gdy dla każdego elementu $\check{\eta}$ z dziedziny τ i dla każdego warunku q z obciążenia do p obrazu $\tau[\check{\eta}]$ tego elementu, jego obraz $\sigma[\check{\eta}]$ w metazbiorze σ pokrywa silniejszy z warunków p, q . Inaczej mówiąc, jeśli przyjmiemy $r = \min(p, q)$, to $\sigma[\check{\eta}]$ zawiera skończony antyłańcuch maksymalny poniżej r lub warunek słabszy od r .

Dowód. Załóżmy, że $\tau \subseteq_p \sigma$. Dla dowolnej gałęzi \mathcal{C} zawierającej warunek p mamy $\tau_{\mathcal{C}} \subset \sigma_{\mathcal{C}}$. To oznacza prawdziwość implikacji: $\check{\eta}_{\mathcal{C}} \in \tau_{\mathcal{C}} \rightarrow \check{\eta}_{\mathcal{C}} \in \sigma_{\mathcal{C}}$ dla dowolnego $\check{\eta} \in \text{dom}(\tau)$. Ustalmy więc jakieś $\check{\eta} \in \text{dom}(\tau)$ i niech $q \in \tau[\check{\eta}] \upharpoonright_p$ (gdyby $\tau[\check{\eta}] \upharpoonright_p = \emptyset$, to $\check{\eta}_{\mathcal{C}} \notin \tau_{\mathcal{C}}$). Niech $r = \min(p, q)$.

Dla dowolnej gałęzi $\mathcal{C}^r \ni r$ jest $\check{\eta}_{\mathcal{C}^r} \in \tau_{\mathcal{C}^r}$, a więc również $\check{\eta}_{\mathcal{C}^r} \in \sigma_{\mathcal{C}^r}$ (gdyż $\tau_{\mathcal{C}^r} \subset \sigma_{\mathcal{C}^r}$). W takim razie $\sigma[\check{\eta}]$ musi zawierać warunek słabszy od r lub antyłańcuch maksymalny poniżej r . Z dziedzicznej skończoności $\sigma[\check{\eta}]$ ten antyłańcuch musi być skończony, więc $\sigma[\check{\eta}]$ pokrywa r , czyli $\sigma[\check{\eta}] \mid \min(p, q)$.

Założmy teraz prawą stronę. Pokażemy, że dla dowolnej gałęzi \mathcal{C} przechodzącej przez p jest $\tau_{\mathcal{C}} \subset \sigma_{\mathcal{C}}$, czyli prawdziwość implikacji $\check{\eta}_{\mathcal{C}} \in \tau_{\mathcal{C}} \rightarrow \check{\eta}_{\mathcal{C}} \in \sigma_{\mathcal{C}}$.

Ustalmy taką gałąź i weźmy jakiś $\check{\eta}_C \in \tau_C$. Jest on interpretacją kanonicznego $\check{\eta} \in \text{dom}(\tau)$. Wybierzmy $q \in \tau[\check{\eta}] \upharpoonright_p$ takie, które leży na gałęzi \mathcal{C} (takie q istnieje, bo $\check{\eta}_C \in \tau_C$). Z założenia $\sigma[\check{\eta}] \upharpoonright_r$, więc $\check{\eta} \in_r \sigma$ (tw. 5.2.2). Ponieważ $r = \min(p, q) \in \mathcal{C}$, to mamy $\check{\eta}_C \in \sigma_C$, a stąd $\tau_C \subset \sigma_C$. Z dowolności $\mathcal{C} \ni p$ otrzymujemy $\tau \subseteq_p \sigma$. \square

WNIOSEK 5.2.6. Jeśli $\tau, \sigma \in \mathfrak{M}\mathfrak{F}^1$, to

$$\tau \subseteq \sigma \quad \leftrightarrow \quad \forall_{\check{\eta} \in \text{dom}(\tau)} \forall_{q \in \tau[\check{\eta}]} \sigma[\check{\eta}] \upharpoonright_q .$$

Dowód. Przyjmujemy $p = 1$ w twierdzeniu 5.2.5. \square

5.2.3 Równość

Lemat 5.2.4 stwierdza, że zawieranie się metazbiorów pierwszego poziomu implikuje zawieranie się ich dziedzin. Dla równości zachodzi podobny związek.

LEMAT 5.2.7. Jeśli $\xi, \eta \in \mathfrak{M}^1$ i $\xi \approx \eta$, to $\text{dom}(\xi) = \text{dom}(\eta)$.

Dowód. Z wniosku 4.2.8 mamy $\xi \approx \eta \leftrightarrow \xi \subseteq \eta \wedge \xi \subseteq \eta$. Stosując dwa razy lemat 5.2.4 dostajemy tezę. \square

Implikacja odwrotna nie zachodzi, podobnie jak dla zawierania, co pokazuje przykład 5.2.2. Jeśli $\xi \approx_p \eta$ dla jakiegoś $p \in \mathbb{T}$, to nie ma żadnego związku pomiędzy ich dziedzinami (w sensie zawierania) poza tym, że muszą mieć niepustą część wspólną (por. przykład 5.2.4).

Twierdzenie 5.2.5 oraz lemat 4.2.7 pozwalają podać charakteryzację relacji równości warunkowej dla dziedzicznie skończonych metazbiorów pierwszego poziomu używając pojęcia pokrywania. Jednak uzyskana w ten sposób formuła jest mało czytelna, dlatego uprościmy ją przy pomocy pojęcia równoważności podzbiorów \mathbb{T} .

TWIERDZENIE 5.2.8. Niech $\tau, \sigma \in \mathfrak{M}\mathfrak{F}^1$, $p \in \mathbb{T}$, a $D = \text{dom}(\tau) \cup \text{dom}(\sigma)$. Wtedy:

$$\tau \approx_p \sigma \quad \leftrightarrow \quad \forall_{\check{\mu} \in D} \tau[\check{\mu}] \parallel_p \sigma[\check{\mu}] .$$

Dowód. Jeśli $\tau = \sigma = \emptyset$, to $D = \emptyset$ i na odwrót. Równoważność jest wtedy spełniona.

Założmy lewą stronę i weźmy $\check{\mu} \in D$. Jeśli $\tau[\check{\mu}] \upharpoonright_p = \emptyset$, to również musi być $\sigma[\check{\mu}] \upharpoonright_p = \emptyset$, bo w przeciwnym przypadku znajdziemy gałąź $\mathcal{C} \ni p$ taką, że $\check{\mu}_C \in \sigma_C$, ale jednocześnie $\check{\mu}_C \notin \tau_C$ (co przeczy założeniu). Jeśli oba $\tau[\check{\mu}] \upharpoonright_p$ i $\sigma[\check{\mu}] \upharpoonright_p$ są puste, to $\tau[\check{\mu}] \parallel_p \sigma[\check{\mu}]$.

Założmy zatem, że $\tau[\check{\mu}] \upharpoonright_p \neq \emptyset$ i wybierzmy $q \in \tau[\check{\mu}] \upharpoonright_p$. Pokażemy, że $\sigma[\check{\mu}] \upharpoonright_p \upharpoonright_{\min(p, q)}$. Dla gałęzi \mathcal{C} przechodzącej przez p i q mamy $\check{\mu}_C \in \tau_C$, i korzystając z założenia o równości warunkowej dostajemy $\check{\mu}_C \in \sigma_C$. Z tego wynika, że musi istnieć $r \in \mathcal{C} \cap \sigma[\check{\mu}] \upharpoonright_p$. Jeśli $r \geq \min(p, q)$, to $\sigma[\check{\mu}] \upharpoonright_p \upharpoonright_{\min(p, q)}$.

W przeciwnym przypadku — gdy $r < \min(p, q)$ — z każdej gałęzi \mathcal{C} zawierającej jednocześnie p i q wybieramy warunek $r_C \leq \min(p, q)$, który należy do $\sigma[\check{\mu}] \upharpoonright_p$. Warunki $r_C \in \mathcal{C} \cap \sigma[\check{\mu}] \upharpoonright_p$ muszą istnieć, bo dla takich gałęzi mamy $\check{\mu}_C \in \tau_C$, a więc i $\check{\mu}_C \in \sigma_C$. Zbiór wszystkich tych warunków zawiera antyłańcuch maksymalny poniżej $\min(p, q)$. Jego skończoność wynika z dziedzicznej skończoności σ . Zatem $\sigma[\check{\mu}] \upharpoonright_p \upharpoonright_{\min(p, q)}$.

Tak samo pokazujemy $\tau[\check{\mu}] \upharpoonright_p \mid \min(p, q)$ dostając $\tau[\check{\mu}] \parallel_p \sigma[\check{\mu}]$. Z dowolności $\check{\mu}$ otrzymujemy prawą stronę.

Założmy teraz, że zbiory $\tau[\check{\mu}]$ i $\sigma[\check{\mu}]$ są równoważne poniżej p dla każdego $\check{\mu} \in D$. Jeśli $D = \emptyset$, to τ, σ są puste, a przez to równe poniżej p . Żeby pokazać $\tau \approx_p \sigma$ dla $D \neq \emptyset$ weźmy dowolną gałąź \mathcal{C} zawierającą p i jakiś zbiór $x \in \tau_{\mathcal{C}}$. Musi on być interpretacją $\check{\mu}_{\mathcal{C}} = x$ jakiegoś $\check{\mu} \in \text{dom}(\tau)$ (fakt 3.3.5). W takim razie istnieje warunek $q \in \tau[\check{\mu}] \upharpoonright_p$ leżący na gałęzi \mathcal{C} . Z równoważności poniżej p zbiorów $\tau[\check{\mu}]$ i $\sigma[\check{\mu}]$ znajdujemy w $\sigma[\check{\mu}] \upharpoonright_p$ warunek r również leżący na \mathcal{C} (powyżej $\min(p, q)$ albo wybieramy z antyłańcucha poniżej). Dzięki temu mamy $\check{\mu}_{\mathcal{C}} \in \sigma_{\mathcal{C}}$, a to daje $\tau_{\mathcal{C}} \subset \sigma_{\mathcal{C}}$. Teraz wybieramy zbiór $y \in \sigma_{\mathcal{C}}$ i powtarzamy tę część dowodu w drugą stronę otrzymując $\sigma_{\mathcal{C}} \subset \tau_{\mathcal{C}}$. Z dowolności \mathcal{C} dostajemy $\tau \approx_p \sigma$. \square

Przyjmując $p = 1$ w twierdzeniu 5.2.8 otrzymujemy jako wniosek warunek na równość dziedzicznie skończonych metazbiory pierwszego poziomu.

WNIOSEK 5.2.9. Jeśli $\tau, \sigma \in \mathfrak{M}\mathfrak{Z}^1$ a $D = \text{dom}(\tau) \cup \text{dom}(\sigma)$, to

$$\tau \approx \sigma \quad \leftrightarrow \quad \forall_{\check{\mu} \in D} \tau[\check{\mu}] \parallel \sigma[\check{\mu}] .$$

5.3 Operacje algebraiczne

W tej sekcji zdefiniujemy operacje algebraiczne dla meta zbiorów: sumę, część wspólną i dopełnienie. Ograniczymy się do metazbiory pierwszego poziomu, dzięki czemu będzie można wyrazić definicje w terminach relacji pomiędzy podzbiorymi drzewa binarnego. Podzbiory te będą obrazami elementów potencjalnych metazbiory będących argumentami operacji.

Taki sposób sformułowania ułatwi stworzenie efektywnych algorytmów realizujących te operacje. Z drugiej strony, ograniczenie się do klasy \mathfrak{M}^1 niesie ze sobą pewne ograniczenia, gdyż zakładamy w ten sposób, że wszystkie elementy potencjalne rozważanych metazbiory zachowują się identycznie w każdej interpretacji.

W dalszym ciągu (sekcja 5.4) pokażemy, że zdefiniowane operacje mają standardowe właściwości i zachowują się zgodnie ze zdrowym rozsądkiem. W sekcji 5.5 pokażemy, że spełniają one aksjomaty algebry Boole'a.

5.3.1 Suma

Suma jest najprostszą operacją na metazbiorych. W przeciwieństwie do pozostałych operacji zdefiniujemy ją w ogólnym przypadku, a nie tylko dla metazbiory pierwszego poziomu.

DEFINICJA 5.3.1. Niech $\tau, \eta \in \mathfrak{M}$ będą dowolne. Sumą metazbiory τ, η , oznaczaną symbolem \cup , nazywamy ich zwykłą sumę mnogościową:

$$\tau \cup \eta = \tau \cup \eta .$$

Oczywiście, $\tau \cup \eta$ jest metazbioryem, o ile τ i η nimi są. Do tego mamy:

FAKT 5.3.1. Dla dowolnych metazbiorów τ, η i $n \in \mathbb{N}$, $n \geq 1$:

$$\begin{aligned} \tau, \eta \in \mathfrak{M}^n &\rightarrow \tau \cup \eta \in \mathfrak{M}^n, \\ \tau, \eta \in \mathfrak{M}\mathfrak{F} &\rightarrow \tau \cup \eta \in \mathfrak{M}\mathfrak{F}, \\ \tau, \eta \in \mathfrak{M}\mathfrak{F}^n &\rightarrow \tau \cup \eta \in \mathfrak{M}\mathfrak{F}^n. \end{aligned}$$

Odnotujmy jeszcze jeden prosty fakt.

FAKT 5.3.2. Jeśli $\tau, \eta \in \mathfrak{M}$, to $\text{dom}(\tau \cup \eta) = \text{dom}(\tau) \cup \text{dom}(\eta)$.

Musimy się upewnić, że tak zdefiniowana suma zachowuje się zgodnie z oczekiwaniami. Trzeba wykazać, że jeśli ξ jest sumą metazbiorów τ i η , to dla każdej gałęzi \mathcal{C} , interpretacja $\xi_{\mathcal{C}}$ zadana przez \mathcal{C} będzie sumą interpretacji składników $\tau_{\mathcal{C}}$ i $\eta_{\mathcal{C}}$.

TWIERDZENIE 5.3.3. Niech τ, η będą dowolnymi metazbiorami, a $\xi = \tau \cup \eta$. Wtedy:

$$\Vdash \xi = \tau \cup \eta.$$

Dowód. Trzeba pokazać, że dla dowolnej gałęzi $\mathcal{C} \subset \mathbb{T}$ zachodzi równość:

$$\xi_{\mathcal{C}} = \tau_{\mathcal{C}} \cup \eta_{\mathcal{C}}.$$

Ustalmy jakąś gałąź \mathcal{C} . Jeśli $x \in \xi_{\mathcal{C}}$, to na mocy faktu 3.3.4 istnieje para $\langle \sigma, p \rangle \in \xi$ taka, że $p \in \mathcal{C}$ i $\sigma_{\mathcal{C}} = x$. Z definicji ξ jest $\langle \sigma, p \rangle \in \tau$ lub $\langle \sigma, p \rangle \in \eta$, a więc $\sigma_{\mathcal{C}} \in \tau_{\mathcal{C}}$ lub $\sigma_{\mathcal{C}} \in \eta_{\mathcal{C}}$, czyli $\sigma_{\mathcal{C}} = x \in \tau_{\mathcal{C}} \cup \eta_{\mathcal{C}}$.

Z kolei, jeśli $x \in \tau_{\mathcal{C}} \cup \eta_{\mathcal{C}}$, to $x \in \tau_{\mathcal{C}}$ lub $x \in \eta_{\mathcal{C}}$. Załóżmy pierwszą wersję. Podobnie jak poprzednio, z faktu 3.3.4 wynika istnienie $\langle \sigma, p \rangle \in \tau$ takiego, że $p \in \mathcal{C}$ i $\sigma_{\mathcal{C}} = x$. Z definicji $\langle \sigma, p \rangle \in \xi$, a stąd mamy $x = \sigma_{\mathcal{C}} \in \xi_{\mathcal{C}}$. \square

Twierdzenie 5.3.3 mówi, że w każdej interpretacji zbiór $(\tau \cup \eta)_{\mathcal{C}}$ jest równy sumie zbiorów $\tau_{\mathcal{C}} \cup \eta_{\mathcal{C}}$, a to oznacza, że suma metazbiorów jest dobrze zdefiniowana.

5.3.2 Część wspólna

Do zdefiniowania części wspólnej metazbiorów pierwszego poziomu potrzebne będą dodatkowe pojęcia.

DEFINICJA 5.3.2. Niech $P, Q \subset \mathbb{T}$ będą dowolnymi podzbiarami drzewa \mathbb{T} . Przez *półspłot* tych zbiorów rozumiemy zbiór:

$$P \times Q = \left\{ p \in P : \exists_{q \in Q} q \geq p \right\}.$$

Półspłot można również oznaczać symbolem \times , którego znaczenie jest następujące:

$$P \times Q = Q \times P = \left\{ q \in Q : \exists_{p \in P} p \geq q \right\}.$$

Łatwo widzieć, że $P \times Q \subset P$. Jeśli $P = \emptyset$ lub $Q = \emptyset$, to wtedy $P \times Q = \emptyset$. Jeśli $r \in P \times Q$ i $\mathcal{C} \subset \mathbb{T}$ jest gałęzią przechodzącą przez r , to $\mathcal{C} \cap Q \neq \emptyset$, czyli taka gałąź zawiera również jakiś element zbioru Q . Taki jest sens półspłotu. Prawdziwa jest więc implikacja (dla dowolnej gałęzi \mathcal{C}):

$$\mathcal{C} \cap (P \times Q) \neq \emptyset \rightarrow \mathcal{C} \cap Q \neq \emptyset \wedge \mathcal{C} \cap P \neq \emptyset. \quad (5.3.1)$$

DEFINICJA 5.3.3. Niech $P, Q \subset \mathbb{T}$ będą dowolnymi podzbiórmi drzewa \mathbb{T} . Splot zbiorów P i Q to zbiór:

$$P \bowtie Q = (P \times Q) \cup (P \rtimes Q) .$$

Bezpośrednio z definicji mamy:

$$P \bowtie Q = \left\{ p \in P : \exists_{q \in Q} q \geq p \right\} \cup \left\{ q \in Q : \exists_{p \in P} p \geq q \right\} . \quad (5.3.2)$$

Jeśli jeden ze zbiorów P, Q jest pusty, to ich splot również.

Niech $r \in P \bowtie Q$ i jednocześnie $r \in \mathcal{C}$, dla pewnej gałęzi \mathcal{C} . Jeśli $r \in P$, to $\mathcal{C} \cap Q \neq \emptyset$, i na odwrót: jeśli $r \in Q$, to $\mathcal{C} \cap P \neq \emptyset$.

Przyjmujemy, że splot i półsplot wiążą tak samo — silniej niż suma i słabiej niż część wspólna zbiorów. Ilustruje to poniższa równość.

$$P \cup Q \bowtie R \cap S = P \cup (Q \bowtie (R \cap S)) . \quad (5.3.3)$$

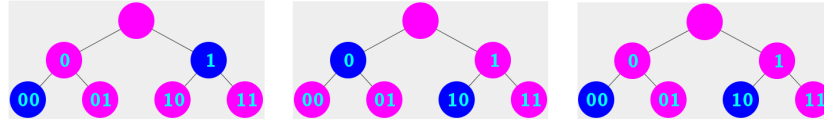
Na wszelki wypadek jednak będziemy unikać notacji, która powodowałyby niejednoznaczność.

Elementami potencjalnymi części wspólnej metazbiorów τ i η będą te meta zbiory, które należą jednocześnie do obu dziedzin $\text{dom}(\tau)$ i $\text{dom}(\eta)$. Jednak stopień należenia do części wspólnej każdego takiego elementu potencjalnego może być inny niż stopnie należenia do argumentów. Oczywiście, nie może być od nich większy.

DEFINICJA 5.3.4. Niech $\tau, \eta \in \mathfrak{M}^\dagger$. Częścią wspólną metazbiorów τ, η nazywamy zbiór:

$$\tau \cap \eta = \{ \langle \xi, p \rangle : \xi \in \text{dom}(\tau) \cap \text{dom}(\eta) \wedge p \in \tau[\xi] \bowtie \eta[\xi] \} .$$

Na rys. 5.3 widzimy część wspólną metazbiorów $\tau = \{ \langle \emptyset, [00] \rangle, \langle \emptyset, [1] \rangle \}$ i $\eta = \{ \langle \emptyset, [0] \rangle, \langle \emptyset, [10] \rangle \}$, która jest równa $\tau \cap \eta = \{ \langle \emptyset, [00] \rangle, \langle \emptyset, [10] \rangle \}$.



Rysunek 5.3: Część wspólna metazbiorów $\{ \emptyset^{[00]}, \emptyset^{[1]} \}$ i $\{ \emptyset^{[0]}, \emptyset^{[10]} \}$

Jak widać, o stopniu przynależności elementu potencjalnego ξ do części wspólnej $\tau \cap \eta$ decyduje sposób jego należenia do obu argumentów τ i η . Ten stopień przynależności $(\tau \cap \eta)[\xi]$ jest zdeterminowany przez wzajemne relacje obrazów $\tau[\xi]$ i $\eta[\xi]$ elementu potencjalnego ξ w danych metazbiorach τ i η . W szczególności może istnieć $\xi \in \text{dom}(\tau) \cap \text{dom}(\eta)$ takie, że $\tau[\xi] \bowtie \eta[\xi] = \emptyset$ (mimo, że $\tau[\xi] \neq \emptyset \neq \eta[\xi]$), i wtedy ξ nie jest elementem potencjalnym części wspólnej ($\xi \notin \text{dom}(\tau \cap \eta)$). Zatem mamy:

FAKT 5.3.4. Jeśli $\tau, \eta \in \mathfrak{M}^\dagger$, to $\text{dom}(\tau \cap \eta) \subset \text{dom}(\tau) \cap \text{dom}(\eta)$.

Jeśli chodzi o obraz części wspólnej, to mamy $\text{ran}(\xi \cap \mu) \subset \text{ran}(\xi) \cup \text{ran}(\mu)$, gdyż dla $\eta \in \text{dom}(\xi \cap \mu)$ jest $(\xi \cap \mu)[\eta] = \xi[\eta] \bowtie \mu[\eta] \subset \xi[\eta] \cup \mu[\eta]$. Z tego wynika następujący fakt.

FAKT 5.3.5. Dla metazbiorów τ i η zachodzą następujące implikacje:

$$\begin{aligned} \tau, \eta \in \mathfrak{M}^\perp &\rightarrow \tau \cap \eta \in \mathfrak{M}^\perp, \\ \tau, \eta \in \mathfrak{M}\mathfrak{F}^\perp &\rightarrow \tau \cap \eta \in \mathfrak{M}\mathfrak{F}^\perp. \end{aligned}$$

Teraz pokażemy, że część wspólna jest dobrze określona.

TWIERDZENIE 5.3.6. Niech $\tau, \eta \in \mathfrak{M}^\perp$ i niech $\xi = \tau \cap \eta$. Wtedy:

$$\Vdash \xi = \tau \cap \eta.$$

Dowód. Trzeba pokazać, że dla dowolnej gałęzi \mathcal{C} zachodzi

$$\xi_{\mathcal{C}} = \tau_{\mathcal{C}} \cap \eta_{\mathcal{C}}.$$

Ustalmy zatem jakąś gałąź \mathcal{C} , i wybierzmy $x \in \xi_{\mathcal{C}}$.

Fakt 3.3.4 gwarantuje istnienie $\sigma \in \text{dom}(\xi)$ i $p \in \xi[\sigma]$ takich, że $p \in \mathcal{C}$ i $\sigma_{\mathcal{C}} = x$, czyli $\langle \sigma, p \rangle \in \xi$. Z definicji części wspólnej mamy $\sigma \in \text{dom}(\tau) \cap \text{dom}(\eta)$ oraz $p \in \tau[\sigma] \cap \eta[\sigma]$. Jeśli $p \in \tau[\sigma]$, to $\langle \sigma, p \rangle \in \tau$ i $\sigma_{\mathcal{C}} \in \tau_{\mathcal{C}}$, oraz $\exists_{q \geq p} q \in \eta[\sigma]$, więc również $\langle \sigma, q \rangle \in \eta$ i $\sigma_{\mathcal{C}} \in \eta_{\mathcal{C}}$. Stąd, ponieważ $x \in \eta_{\mathcal{C}}$ i $x \in \tau_{\mathcal{C}}$, mamy $x \in \tau_{\mathcal{C}} \cap \eta_{\mathcal{C}}$. W przeciwnym przypadku, jeśli $p \notin \tau[\sigma]$, to musi być $p \in \eta[\sigma]$. Wtedy $\langle \sigma, p \rangle \in \eta$ i $\sigma_{\mathcal{C}} \in \eta_{\mathcal{C}}$, oraz $\exists_{q \geq p} q \in \tau[\sigma]$, więc również $\langle \sigma, q \rangle \in \tau$ i $\sigma_{\mathcal{C}} \in \tau_{\mathcal{C}}$. Podobnie jak poprzednio otrzymujemy $x = \sigma_{\mathcal{C}} \in \tau_{\mathcal{C}} \cap \eta_{\mathcal{C}}$.

Niech teraz $x \in \tau_{\mathcal{C}} \cap \eta_{\mathcal{C}}$. Fakty 3.3.4 i 3.3.3 implikują istnienie jedynego $\sigma \in \text{dom}(\tau) \cap \text{dom}(\eta)$ takiego, że $\sigma_{\mathcal{C}} = x$. Istnieją również $p, q \in \mathcal{C}$ takie, że $\langle \sigma, p \rangle \in \tau$ i $\langle \sigma, q \rangle \in \eta$. Jeśli $p \leq q$, to $p \in \tau[\sigma] \cap \eta[\sigma]$. W przeciwnym przypadku $q \in \tau[\sigma] \cap \eta[\sigma]$. Zatem istnieje $r \in \tau[\sigma] \cap \eta[\sigma]$ ($r = p$ lub $r = q$). Stąd $\langle \sigma, r \rangle \in \xi$, a ponieważ $r \in \mathcal{C}$, to $x = \sigma_{\mathcal{C}} \in \xi_{\mathcal{C}}$. \square

Pokazaliśmy, że dla dowolnej gałęzi \mathcal{C} jest $(\tau \cap \eta)_{\mathcal{C}} = \tau_{\mathcal{C}} \cap \eta_{\mathcal{C}}$, co wraz z faktem 5.3.5 uzasadnia poprawność definicji.

5.3.3 Różnica

Definicja różnicy metazbiorów jest bardziej złożona niż definicje sumy i części wspólnej. W przeciwieństwie do różnicy zwykłych zbiorów, w przypadku metazbiorów różnica τ i η będzie zawierać nie tylko te elementy z τ , które nie znajdują się w η , ale również takie, które częściowo występują zarówno w τ , jak i w η . W szczególności, jeśli w τ „jest więcej” elementu σ niż w η , to różnica τ i η powinna zawierać pewną ilość elementu σ . Żeby wyrazić te niuanse będziemy potrzebowali kilku dodatkowych pojęć. Najpierw wprowadzimy pewne użyteczne oznaczenia i wymienimy ich najważniejsze właściwości.

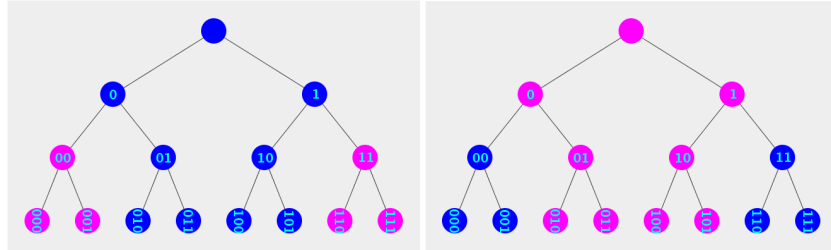
Niech $P \subset \mathbb{T}$ będzie zbiorem warunków w drzewie \mathbb{T} . Przez P^\top będziemy oznaczać zbiór warunków porównywalnych z elementami z P :

$$P^\top = \left\{ q \in \mathbb{T} : \exists_{p \in P} p \top q \right\}. \quad (5.3.4)$$

Podobnie, przez P^\perp będziemy oznaczać zbiór warunków nieporównywalnych z żadnym elementem z P :

$$P^\perp = \left\{ q \in \mathbb{T} : \forall_{p \in P} p \perp q \right\}. \quad (5.3.5)$$

Przyjmujemy, że jeśli $P = \emptyset$, to $P^\top = \emptyset$ i $P^\perp = \mathbb{T}$. Rysunek 5.4 pokazuje przykłady zbiorów P^\top i P^\perp dla $P = \{[01], [10]\}$ (widoczne są tylko 4 poziomy drzewa).



Rysunek 5.4: Zbiory P^\top i P^\perp dla $P = \{[01], [10]\}$

Elementy zbioru P^\top leżą na gałęziach wyznaczonych przez elementy z P . Żaden element zbioru P^\perp nie leży na jednej gałęzi z jakimkolwiek elementem z P . Oczywiście, $P \subset P^\top$. Z kolei, jeśli $\mathbb{1} \in P$, to $P^\top = \mathbb{T}$ i $P^\perp = \emptyset$. Natomiast jeśli $p \neq \mathbb{1}$, to $\{p\}^\top$ składa się z poddrzewa o korzeniu p i gałęzi przechodzącej przez p , a więc zachodzi równość $\{p\}^\top = \mathbb{T} \upharpoonright_p$. Ponadto:

FAKT 5.3.7. Dla $P \subset \mathbb{T}$ spełnione są równości:

$$\begin{aligned} P^\top \cup P^\perp &= \mathbb{T} , \\ P^\top \cap P^\perp &= \emptyset . \end{aligned}$$

Niech $P = \{[11]\}$. P^\top składa się z poddrzewa o korzeniu $[11]$ oraz elementu $[1]$ i korzenia $\mathbb{1}$. P^\perp składa się z dwóch poddrzew o korzeniach $[0]$ i $[10]$, czyli z warunków silniejszych od $[0]$ i $[10]$. Zauważmy, że warunki $[0]$, $[10]$ i $[11]$ tworzą skończony antyłańcuch maksymalny.

Niech $P \subset \mathbb{T}$ będzie jakimś zbiorem warunków. Przez $\max(P)$ oznaczymy zbiór elementów maksymalnych w P . Zatem

$$p \in \max(P) \quad \leftrightarrow \quad p \in P \wedge \forall_{q \in P} (q \geq p \rightarrow q = p) . \quad (5.3.6)$$

Przyjmujemy, że $\max(\emptyset) = \emptyset$, oraz $\max(\mathbb{T}) = \{\mathbb{1}\}$. Interesującą nas cechą zbioru $\max(P)$ jest to, że każdy element zbioru P jest porównywalny z jakimś elementem z $\max(P)$. Co więcej, jest od niego silniejszy. W przytoczonym powyżej przykładzie $\max(P^\perp) = \{[0], [10]\}$ (por. też rys. 5.5).

Udowodnimy teraz dwie ważne właściwości podzbiorów drzewa \mathbb{T} konieczne do zdefiniowania różnicy metazbiorów.

LEMAT 5.3.8. Niech $P \subset \mathbb{T}$. Zbiór $\max(P)$ elementów maksymalnych z P jest antyłańcuchem maksymalnym w P .

Dowód. Elementy zbioru $\max(P)$ są nieporównywalne, więc jest on antyłańcuchem. Ponadto, każdy element z P jest porównywalny z jakimś elementem z $\max(P)$, więc jest on maksymalny w P . \square

LEMAT 5.3.9. Jeśli P jest skończonym podzbiorem \mathbb{T} , to zbiór $\max(P^\perp)$ jest skończonym antyłańcuchem maksymalnym w P^\perp .

Dowód. To, że $\max(P^\perp)$ jest antyłańcuchem maksymalnym w P^\perp wynika z lematu 5.3.8. Pokażemy, że jeśli P jest skończony, to $\max(P^\perp)$ też jest skończony.

Oznaczmy symbolem p^\leq zbiór warunków silniejszych od danego $p \in \mathbb{T}$, czyli poddrzewo o korzeniu p : $p^\leq = \{q \in \mathbb{T} : q \leq p\}$. Zauważmy, że $\max(p^\leq) = \{p\}$, oraz:

$$\max(q^\leq \cup r^\leq) = \begin{cases} \{\max(q, r)\} & \text{gdy } q \top r, \\ \{q, r\} & \text{gdy } q \perp r. \end{cases} \quad (5.3.7)$$

Powyższy wzór uogólnić na dowolną skończoną liczbę składników.

Dla $s \in \mathbb{T}$ zbiór elementów nieporównywalnych z s , czyli $\{s\}^\perp$, jest skończoną sumą poddrzew:

$$\{s\}^\perp = s_1^\leq \cup \dots \cup s_n^\leq, \quad (5.3.8)$$

gdzie n jest numerem poziomu drzewa \mathbb{T} zawierającym s , a s_i jest pewnym warunkiem z poziomu i . Rzeczywiście, jeśli np. $s = [0]$, to $\{s\}^\perp = \{[0]\}^\perp = [1]^\leq$, a $[1] \in \mathbb{T}_1$. Z kolei $\{[00]\}^\perp = [01]^\leq \cup [1]^\leq$. Ogólnie, dla $s \in \mathbb{T}_n$ jako s_n wybieramy bliźniaka węzła s , jako s_{n-1} wybieramy bliźniaka wspólnego rodzica s i s_n , i tak dalej, aż do poziomu pierwszego, z którego jako s_1 wybieramy węzeł nieporównywalny z żadnym z s_n, s_{n-1}, \dots, s_2 . Korzystając z (5.3.8) widzimy, że:

$$\max(\{s\}^\perp) = \max(s_1^\leq \cup \dots \cup s_n^\leq) \subset \{s_1, \dots, s_n\} \quad (5.3.9)$$

jest zbiorem skończonym. Dalej, zauważmy, że dla $Q, R \subset \mathbb{T}$ zachodzi związek:

$$(Q \cup R)^\perp = Q^\perp \cap R^\perp.$$

W takim razie dla $P = \{p_1, \dots, p_m\}$ mamy:

$$P^\perp = \{p_1, \dots, p_m\}^\perp = \{p_1\}^\perp \cap \dots \cap \{p_m\}^\perp = \bigcap_{i=1}^m \{p_i\}^\perp.$$

Rozwijając poszczególne $\{p_i\}^\perp$ w sumy dzięki (5.3.8) otrzymujemy:

$$P^\perp = \bigcap_{i=1}^m \{p_i\}^\perp = \bigcap_{i=1}^m \bigcup_{j=1}^{n_i} p_{ij}^\leq. \quad (5.3.10)$$

Wymnażając sumy dostaniemy wyrażenie postaci: $P^\perp = \bigcup_{k=1}^l \bigcap_{i=1}^m p_{ij_{ik}}^\leq$ dla $l = n_1 \cdot \dots \cdot n_m$, przy czym indeks $j_{ik} \leq n_i$ zależy od i oraz k . Biorąc pod uwagę, że:

$$q^\leq \cap r^\leq = \begin{cases} \emptyset & \text{gdy } q \perp r, \\ q^\leq & \text{gdy } q \leq r, \\ r^\leq & \text{gdy } q \geq r, \end{cases}$$

i uogólniając powyższą zależność na dowolną skończoną liczbę czynników dostaniemy:

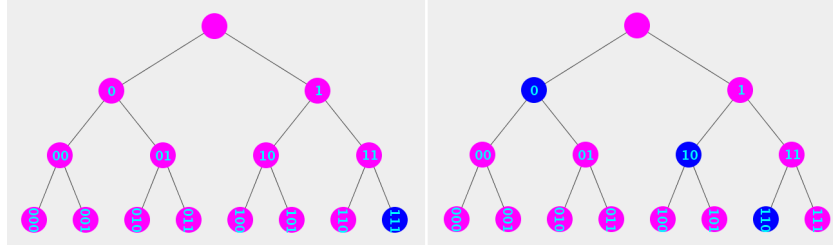
$$\begin{aligned} P^\perp &= \bigcup_{k=1}^l \bigcap_{i=1}^m p_{ij_{ik}}^\leq, \\ &= \bigcup_{k=1}^l P_k, \quad \text{gdzie } P_k = \begin{cases} p_{i_k j_{i_k}}^\leq & \text{dla } p_{i_k j_{i_k}}^\leq = \bigcap_{i=1}^m p_{ij_{ik}}^\leq, \\ \emptyset & \text{gdy } \bigcap_{i=1}^m p_{ij_{ik}}^\leq = \emptyset. \end{cases} \end{aligned}$$

A stąd i z (5.3.9) otrzymujemy, że $\max(P^\perp)$ jest zbiorem skończonym, bo

$$\max(P^\perp) \subset \{p_{ij}\}_{i \leq m, j \leq n_i} ,$$

gdzie każde $p_{ij} = \max(p_{ij}^{\leq})$ jest określone w (5.3.10). \square

DEFINICJA 5.3.5. Dla $P \subset \mathbb{T}$, zbiór $\max(P^\perp)$ będziemy nazywać *bazą dopełnienia zbioru P* .



Rysunek 5.5: Zbiór $P = \{[111]\}$ i jego baza dopełnienia

Jak wyjaśniono na początku tej sekcji, różnica σ metazbiorów τ i η może zawierać elementy potencjalne, które po trochu należą do τ i η jednocześnie. Dlatego zanim będziemy mogli zdefiniować różnicę, musimy określić, co to dokładnie znaczy, w jakich stopniach elementy potencjalne różnicy należą do równocześnie do argumentów tej operacji. Pojęcie brzegu, które wprowadzamy poniżej określa właśnie to: ilość każdego elementu potencjalnego $\xi \in \text{dom}(\tau) \cap \text{dom}(\eta)$, która zostaje w różnicy po odjęciu metazbioru η .

Jeśli τ jest metazbiorem kanonicznym, to czym powinna być różnica σ metazbiorów τ i η ? Każdy element potencjalny w różnicy σ powinien do niej należeć w takim stopniu, w jakim nie należy do η , w jakim go brakuje w η . Wtedy, jeśli dodamy η i σ , to dostaniemy τ albo metazbiór, który jest jemu równy, którego elementy potencjalne należą do niego w pełni, jak w metazbiorze kanonicznym. Podobne intuicje dotyczą ogólnego przypadku, w którym elementy potencjalne mogą należeć w stopniu mniejszym niż $\mathbb{1}$.

DEFINICJA 5.3.6. Niech $\tau, \eta \in \mathfrak{M}^1$. *Brzegiem metazbioru η w meta zbiorze τ* nazywamy meta zbiór:

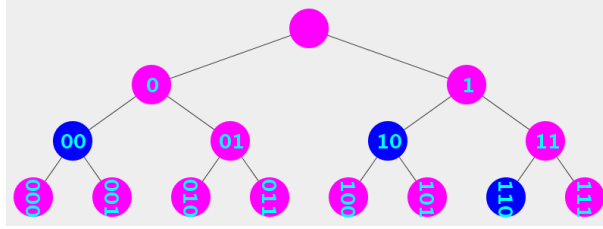
$$\tilde{\eta}^\tau = \{ \langle \xi, p \rangle : \xi \in \text{dom}(\tau) \cap \text{dom}(\eta) \wedge p \in \tau[\xi] \cap \eta[\xi]^\perp \cup \max(\eta[\xi]^\perp) \times \tau[\xi] \} .$$

Jeśli $\text{dom}(\tau) \cap \text{dom}(\eta) = \emptyset$, to oczywiście $\tilde{\eta}^\tau = \emptyset$. Jeśli $\xi \in \text{dom}(\tau) \cap \text{dom}(\eta)$, to $\tilde{\eta}^\tau[\xi]$ składa się z:

- tych warunków nieporównywalnych z warunkami z $\eta[\xi]$, które należą do $\tau[\xi]$, czyli z elementów $\tau[\xi] \cap \eta[\xi]^\perp$,
- oraz z tych warunków z bazy dopełnienia obrazu $\eta[\xi]$, które mają nad sobą jakiś element z $\tau[\xi]$, czyli należą do pólplotu $\max(\eta[\xi]^\perp) \times \tau[\xi]$.

PRZYKŁAD 5.3.1: Dla uzasadnienia i wyjaśnienia definicji rozważmy metazbiory $\tau = \{ \langle \check{\sigma}, p \rangle \}$ i $\eta = \{ \langle \check{\sigma}, q \rangle \}$ dla jakiegoś kanonicznego $\check{\sigma}$, oraz $p = [1]$ i $q = [11]$. Metazbiór $\check{\sigma}$ należy do τ „w większym stopniu” niż do η , bo $\check{\sigma} \in_p \tau$, $\check{\sigma} \in_q \eta$

i $q \leq p$. Inaczej mówiąc, w każdej interpretacji, w której $\check{\sigma}_C \in \eta_C$ jest również $\check{\sigma}_C \in \tau_C$. Chcemy tak zdefiniować brzeg η w τ , by w tej sytuacji był on niepusty i w interpretacjach zachowywał się jak różnica tych zbiorów. Dokładniej, dla gałęzi C takich, że zachodzi $\check{\sigma}_C \in \eta_C$ i $\check{\sigma}_C \in \tau_C$ lub też zachodzi $\check{\sigma}_C \notin \eta_C$ i $\check{\sigma}_C \notin \tau_C$, interpretacja brzegu powinna być pusta. Natomiast dla C takich, że $\check{\sigma}_C \notin \eta_C$ i $\check{\sigma}_C \in \tau_C$, interpretacja brzegu powinna zawierać $\check{\sigma}_C$. A to oznacza dokładnie tyle, że $\tilde{\eta}^\tau[\check{\sigma}]$ musi rozpinać $[10]$, czyli zawierać skończony antylańcuch maksymalny poniżej $[10]$. I tak właśnie ujmujemy brzeg η w τ definicja 5.3.6. Zobaczmy: $\tau[\check{\sigma}] \cap \eta[\check{\sigma}]^\perp = \emptyset$, bo $\tau[\check{\sigma}] \subset \eta[\check{\sigma}]^\top$ (p jest porównywalne z q). Z kolei $\max(\eta[\check{\sigma}]^\perp) = \{[0], [10]\}$. Ale tylko element $r = [10]$ ma nad sobą jakiś element z $\tau[\check{\sigma}]$, którym jest $p = [1]$. Stąd półspłot $\max(\eta[\check{\sigma}]^\perp) \times \tau[\check{\sigma}]$ zawiera tylko warunek r i ostatecznie $\tilde{\eta}^\tau = \{\langle \check{\sigma}, r \rangle\}$.



Rysunek 5.6: Brzeg $\eta = \{\langle \emptyset, [111] \rangle\}$ w $\tau = \{\langle \emptyset, [1] \rangle, \langle \emptyset, [00] \rangle\}$.

Rysunek 5.6 pokazuje brzeg $\eta = \{\langle \emptyset, [111] \rangle\}$ w $\tau = \{\langle \emptyset, [1] \rangle, \langle \emptyset, [00] \rangle\}$, który jest równy $\tilde{\eta}^\tau = \{\langle \emptyset, [00] \rangle, \langle \emptyset, [10] \rangle, \langle \emptyset, [110] \rangle\}$. Rzeczywiście, warunek $[00]$ jest w obrazie $\tilde{\eta}^\tau[\emptyset]$, ponieważ należy on do obrazu $\tau[\emptyset]$ i jest nieporównywalny z żadnym elementem obrazu $\eta[\emptyset]$, czyli należy do $[00] \in \tau[\emptyset] \cap \eta[\emptyset]^\perp$. Z kolei pozostałe warunki $[10]$ i $[110]$ znajdują się tam, ponieważ należą do bazy dopełnienia zbioru $\{[111]\}$ (por. rys. 5.5) i jednocześnie mają nad sobą warunek $[1]$ z $\tau[\emptyset]$, a więc należą do $\max(\eta[\emptyset]^\perp) \times \tau[\emptyset]$.

FAKT 5.3.10. Jeśli $\tau, \eta \in \mathfrak{M}^\perp$, to $\text{dom}(\tilde{\eta}^\tau) \subset \text{dom}(\tau) \cap \text{dom}(\eta)$.

Prawdziwość powyższego faktu wynika bezpośrednio z definicji. Natomiast warto zwrócić uwagę na to, że nie można postawić równości zamiast zawierania, gdyż dla pewnych $\xi \in \text{dom}(\tau) \cap \text{dom}(\eta)$ może być jednocześnie $\tau[\xi] \cap \eta[\xi]^\perp = \emptyset$ i $\max(\eta[\xi]^\perp) \times \tau[\xi] = \emptyset$. Wtedy $\xi \notin \text{dom}(\tilde{\eta}^\tau)$.

PRZYKŁAD 5.3.2: Jeśli $\text{dom}(\tau) = \text{dom}(\eta)$, to może być $\tilde{\eta}^\tau = \emptyset$ (np. $\tilde{\tau}^\tau = \emptyset$), a może być również $\tilde{\eta}^\tau = \tau$. Rozważmy dla przykładu $\tau = \{\langle \check{\sigma}, [0] \rangle\}$ oraz $\eta = \{\langle \check{\sigma}, [1] \rangle\}$ dla jakiegoś kanonicznego $\check{\sigma}$. Mamy $\text{dom}(\tau) = \text{dom}(\eta) = \{\check{\sigma}\}$, oraz $\tau[\check{\sigma}] = \{[0]\}$ i $\eta[\check{\sigma}] = \{[1]\}$. Łatwo widzieć, że $\eta[\check{\sigma}]^\top$ składa się z korzenia $\mathbb{1}$ oraz poddrzewa o korzeniu $[1]$, natomiast $\eta[\check{\sigma}]^\perp$ składa się z poddrzewa o korzeniu $[0]$. Zatem $\tau[\check{\sigma}] \cap \eta[\check{\sigma}]^\perp = \{[0]\}$. W tym przypadku również $\max(\eta[\check{\sigma}]^\perp) = \{[0]\}$, a więc $\max(\eta[\check{\sigma}]^\perp) \times \tau[\check{\sigma}] = \{[0]\}$. Z tego wynika, że $\tilde{\eta}^\tau = \tau$, bo $\tilde{\eta}^\tau[\check{\sigma}] = \{[0]\} = \tau[\check{\sigma}]$. Oczywiście, nigdy nie może być $\tilde{\eta}^\tau = \eta$ (wyłączając przypadek $\tau = \eta = \emptyset$), bo dla $\xi \in \text{dom}(\tilde{\eta}^\tau) \cap \text{dom}(\eta)$ zawsze jest $\tilde{\eta}^\tau[\xi] \cap \eta[\xi] = \emptyset$, gdyż $\tilde{\eta}^\tau[\xi] \subset \eta[\xi]^\perp$.

LEMAT 5.3.11. Dla dowolnych metazbiorów:

$$\begin{aligned} \tau, \eta \in \mathfrak{M}^1 &\rightarrow \tilde{\eta}^\tau \in \mathfrak{M}^1, \\ \tau, \eta \in \mathfrak{M}\mathfrak{F}^1 &\rightarrow \tilde{\eta}^\tau \in \mathfrak{M}\mathfrak{F}^1. \end{aligned}$$

Dowód. Pierwsza implikacja wynika bezpośrednio z definicji. Żeby udowodnić drugą należy pokazać, że dla $\xi \in \text{dom}(\tau) \cap \text{dom}(\eta)$, zbiory $\max(\eta[\xi]^\perp) \times \tau[\xi]$ i $\tau[\xi] \cap \eta[\xi]^\perp$ są skończone. Pierwszy z nich jest skończony, bo jest zawarty w $\max(\eta[\xi]^\perp)$, który jest skończony na mocy założenia, że $\eta \in \mathfrak{M}\mathfrak{F}^1$ i lematu 5.3.9. Skończoność drugiego zbioru wynika z założenia, że $\tau \in \mathfrak{M}\mathfrak{F}^1$, bowiem $\tau[\xi] \cap \eta[\xi]^\perp \subset \tau[\xi]$. \square

Jeśli $\text{dom}(\tau) \subset \text{dom}(\eta)$, to brzeg $\tilde{\eta}^\tau$ jest różnicą τ i η . W ogólnym przypadku trzeba do $\tilde{\eta}^\tau$ coś dodać, aby uzyskać różnicę.

DEFINICJA 5.3.7. Niech $\tau, \eta \in \mathfrak{M}^1$. Różnicą metazbiorów τ i η nazywamy zbiór:

$$\tau \asymp \eta = \tau \upharpoonright_{\text{dom}(\tau) \setminus \text{dom}(\eta)} \cup \tilde{\eta}^\tau.$$

Wyrażenie $\tau \upharpoonright_{\text{dom}(\tau) \setminus \text{dom}(\eta)}$ oznacza obcięcie relacji, którą jest metazbiór τ , do zbioru $\text{dom}(\tau) \setminus \text{dom}(\eta)$, tzn. $\text{dom}(\tau \upharpoonright_{\text{dom}(\tau) \setminus \text{dom}(\eta)}) = \text{dom}(\tau) \setminus \text{dom}(\eta)$.

Przyjrzyjmy się tej definicji. Oczywiście, jeśli $\text{dom}(\tau) \cap \text{dom}(\eta) = \emptyset$, to $\tau \asymp \eta = \tau$. Łatwo zauważyć, że $\tau \asymp \tau = \emptyset$. Istotnie, $\tau \upharpoonright_{\text{dom}(\tau) \setminus \text{dom}(\tau)} = \emptyset$ i $\tilde{\tau}^\tau = \emptyset$. Jeśli $\text{dom}(\tau) = \text{dom}(\eta)$, to pierwszy składnik sumy jest pusty i wtedy $\tau \asymp \eta = \tilde{\eta}^\tau$. W takim przypadku może być $\tau \asymp \eta = \emptyset$ nawet, jeśli $\tau \neq \eta$.

Niech $\sigma = \tau \asymp \eta$. Jeśli $\xi \notin \text{dom}(\tau)$, to $\xi \notin \text{dom}(\sigma)$ niezależnie od tego, czy $\xi \in \text{dom}(\eta)$, czy też nie. Jeśli $\xi \in \text{dom}(\tau)$ i $\xi \notin \text{dom}(\eta)$, to $\xi \in \text{dom}(\sigma)$ zawsze. Jeżeli $\xi \in \text{dom}(\tau) \cap \text{dom}(\eta)$, to $\xi \in \text{dom}(\sigma)$, gdy $\xi \in \text{dom}(\tilde{\eta}^\tau)$, czyli przynajmniej jeden ze zbiorów $\tau[\xi] \cap \eta[\xi]^\perp$, $\max(\eta[\xi]^\perp) \times \tau[\xi]$ jest niepusty. Z tego i z faktu 5.3.10 wynika:

FAKT 5.3.12. Jeśli $\tau, \eta \in \mathfrak{M}^1$, to $\text{dom}(\tau \asymp \eta) \subset \text{dom}(\tau)$.

Z lematu 5.3.11 wynika następujący, ważny fakt.

FAKT 5.3.13. Dla dowolnych metazbiorów τ, η :

$$\begin{aligned} \tau, \eta \in \mathfrak{M}^1 &\rightarrow \tau \asymp \eta \in \mathfrak{M}^1, \\ \tau, \eta \in \mathfrak{M}\mathfrak{F}^1 &\rightarrow \tau \asymp \eta \in \mathfrak{M}\mathfrak{F}^1. \end{aligned}$$

Teraz pokażemy, że różnica metazbiorów jest dobrze zdefiniowana. To znaczy, że każda interpretacja metazbioru $\tau \asymp \eta$ jest różnicą interpretacji metazbiorów τ i η .

TWIERDZENIE 5.3.14. Niech $\tau, \eta \in \mathfrak{M}^1$ i $\xi = \tau \asymp \eta$. Wtedy:

$$\Vdash \xi = \tau \setminus \eta.$$

Dowód. Trzeba pokazać, że dla dowolnej gałęzi $\mathcal{C} \subset \mathbb{T}$ zachodzi:

$$\xi_{\mathcal{C}} = \tau_{\mathcal{C}} \setminus \eta_{\mathcal{C}}.$$

Ustalmy jakąś gałąź \mathcal{C} i niech $x \in \xi_{\mathcal{C}}$. Istnieje $\langle \check{\sigma}, p \rangle \in \xi$ takie, że $p \in \mathcal{C}$ i $\check{\sigma}_{\mathcal{C}} = x$. Możliwe są dwa przypadki: $\check{\sigma} \in \text{dom}(\eta)$ lub $\check{\sigma} \notin \text{dom}(\eta)$. Jeśli $\check{\sigma} \notin \text{dom}(\eta)$, to

zgodnie z definicją $\check{\sigma} \in \text{dom}(\tau)$ i $\langle \check{\sigma}, p \rangle \in \tau$, a stąd $\check{\sigma}_C \in \tau_C$. Ponieważ $\check{\sigma} \notin \text{dom}(\eta)$ i $\eta \in \mathfrak{M}^1$, to $\check{\sigma}_C \notin \eta_C$ i nie może istnieć $\check{\mu} \in \text{dom}(\eta)$ takie, że $\check{\mu}_C = \check{\sigma}_C = x$. W konsekwencji $x = \check{\sigma}_C \in \tau_C \setminus \eta_C$. Jeśli $\check{\sigma} \in \text{dom}(\eta)$, to również $\check{\sigma} \in \text{dom}(\tau)$, bo $\check{\sigma} \in \text{dom}(\xi)$ i w tym przypadku $\check{\sigma} \in \text{dom}(\tilde{\eta}^\tau)$. Zgodnie z definicją $\langle \check{\sigma}, p \rangle \in \tilde{\eta}^\tau$ i $p \in \tau[\check{\sigma}] \cap \eta[\check{\sigma}]^\perp$ lub $p \in \max(\eta[\check{\sigma}]^\perp) \times \tau[\check{\sigma}]$. W pierwszym przypadku $\langle \check{\sigma}, p \rangle \in \tau$ i nie istnieje $q \in \mathcal{C}$ takie, że $q \in \eta[\check{\sigma}]$, więc $\langle \check{\sigma}, q \rangle \notin \eta$ dla $q \in \mathcal{C}$. To oznacza, że $\check{\sigma}_C \in \tau_C$ i $\check{\sigma}_C \notin \eta_C$, więc $x = \check{\sigma}_C \in \tau_C \setminus \eta_C$. W drugim przypadku $p \in \eta[\check{\sigma}]^\perp$, więc nie istnieje $q \in \mathcal{C}$ takie, że $q \in \eta[\check{\sigma}]$, a stąd $\langle \check{\sigma}, q \rangle \notin \eta$ dla $q \in \mathcal{C}$, i w konsekwencji $\check{\sigma}_C \notin \eta_C$. Z definicji półspłotu wynika, że istnieje $r \in \tau[\check{\sigma}]$ takie, że $p \leq r$. Ponieważ $p \in \mathcal{C}$, to również $r \in \mathcal{C}$, a stąd $\check{\sigma}_C \in \tau_C$. Zatem również w drugim przypadku mamy $x = \check{\sigma}_C \in \tau_C \setminus \eta_C$, co ostatecznie dowodzi, że $\xi_C \subset \tau_C \setminus \eta_C$ dla dowolnej gałęzi \mathcal{C} .

Weźmy teraz $x \in \tau_C \setminus \eta_C$. Istnieje $\check{\sigma} \in \text{dom}(\tau)$ takie, że $\check{\sigma}_C = x$. Istnieje również $p \in \mathcal{C}$ takie, że $\langle \check{\sigma}, p \rangle \in \tau$. Jeśli $\check{\sigma} \notin \text{dom}(\eta)$, to (z definicji różnicy) $\langle \check{\sigma}, p \rangle \in \xi$ i $x = \check{\sigma}_C \in \xi_C$. W przeciwnym przypadku $\check{\sigma} \in \text{dom}(\eta) \cap \text{dom}(\tau)$ i jeśli p nie jest porównywalne z żadnym warunkiem z $\eta[\check{\sigma}]$, to $p \in \eta[\check{\sigma}]^\perp$. Ponieważ $p \in \tau[\check{\sigma}]$, to $p \in \tau[\check{\sigma}] \cap \eta[\check{\sigma}]^\perp$. Z definicji brzegu mamy więc $p \in \xi[\check{\sigma}]$ i $\langle p, \check{\sigma} \rangle \in \xi$, a stąd $x = \check{\sigma}_C \in \xi_C$. Jeśli p jest porównywalne z jakimś warunkiem $q \in \eta[\check{\sigma}]$, to $p \in \eta[\check{\sigma}]^\top$. Do tego $q < p$ i $q \notin \mathcal{C}$, bo $\check{\sigma}_C \notin \eta_C$, więc nie istnieje $q \in \mathcal{C}$ (a tym bardziej $q \geq p$) takie, że $\langle \check{\sigma}, q \rangle \in \eta$. Niech r_{min} będzie najsilniejszym warunkiem o tej właściwości co p , tzn. należącym do $\mathcal{C} \cap \eta[\check{\sigma}]^\top$. Taki warunek istnieje, bo $\eta[\check{\sigma}] \cap \mathcal{C} = \emptyset$, więc świadkami tacy jak q leżą poza \mathcal{C} , a $\eta[\check{\sigma}]$ jest skończony. Niech r będzie następnikiem r_{min} należącym do \mathcal{C} . Jest $r \in \max(\eta[\check{\sigma}]^\perp)$, bo $r \in \eta[\check{\sigma}]^\perp$, a $r_{min} \in \eta[\check{\sigma}]^\top$. Oprócz tego $r \leq p$, więc $r \in \max(\eta[\check{\sigma}]^\perp) \times \tau[\check{\sigma}]$. Z definicji brzegu $r \in \xi[\check{\sigma}]$ i $\langle r, \check{\sigma} \rangle \in \xi$, a stąd $x = \check{\sigma}_C \in \xi_C$. \square

5.4 Właściwości relacji i operacji

Przedstawimy tu kilka prostych właściwości relacji i operacji dla metazbiorów, które będą potrzebne w dalszej części.

Suma metazbiorów pierwszego poziomu $\tau \cup \eta$ jest najmniejszym (w sensie metazawierania \subseteq) metazbiorem zawierającym jednocześnie τ i η .

LEMAT 5.4.1. Jeśli $\tau, \gamma, \delta \in \mathfrak{M}\mathfrak{F}^1$, oraz $\tau \subseteq \delta$ i $\gamma \subseteq \delta$, to $\tau \cup \gamma \subseteq \delta$.

Dowód. Oznaczmy $\xi = \tau \cup \gamma$. Z założenia i wniosku 5.2.6 mamy:

$$\bigvee_{\check{\sigma} \in \text{dom}(\tau)} \bigvee_{p \in \tau[\check{\sigma}]} \delta[\check{\sigma}] \mid p, \quad \text{oraz} \quad \bigvee_{\check{\eta} \in \text{dom}(\gamma)} \bigvee_{q \in \gamma[\check{\eta}]} \delta[\check{\eta}] \mid q.$$

Chcemy pokazać $\bigvee_{\check{\mu} \in \text{dom}(\xi)} \bigvee_{r \in \xi[\check{\mu}]} \delta[\check{\mu}] \mid r$. Jeśli $\check{\mu} \in \text{dom}(\xi)$, to $\check{\mu} \in \text{dom}(\tau)$ lub $\check{\mu} \in \text{dom}(\gamma)$ (fakt 5.3.2). W pierwszym przypadku, jeśli $r \in \tau[\check{\mu}]$, to $\delta[\check{\mu}] \mid r$. Analogicznie w drugim przypadku. \square

Dla dowolnego $P \subset \mathbb{T}$ jest oczywiste, że $P \bowtie P = P$. Stąd otrzymujemy następujący fakt (przyjmując $P = \mu[\xi]$ dla $\xi \in \text{dom}(\mu)$).

FAKT 5.4.2. Jeśli $\mu \in \mathfrak{M}^1$, to $\mu \circ \mu = \mu$.

Niech $p = [0]$, $q = [00]$, $r = [000]$ będą elementami drzewa \mathbb{T} , oraz niech $P = \{p, r\}$ i $Q = \{q\}$. Widzimy, że $P \bowtie Q = \{q, r\}$, a stąd $P \bowtie Q \not\subseteq P$ i $P \bowtie Q \not\subseteq Q$. Jeśli $\check{\sigma}$ jest jakimś metazbiorem kanonicznym, to dla $\tau = \{\check{\sigma}^p, \check{\sigma}^r\}$

i $\eta = \{\check{\sigma}^q\}$ mamy $\tau \circ \eta \not\subseteq \tau$ i $\tau \circ \eta \not\subseteq \eta$. Jednak zawieranie zdefiniowane dla metazbiorów zachowuje się poprawnie.

LEMAT 5.4.3. Jeśli $\mu, \tau \in \mathfrak{M}\mathfrak{F}^1$, to $\mu \circ \tau \subseteq \tau$.

Dowód. Trzeba pokazać, że dla $\xi \in \text{dom}(\mu \circ \tau)$ i dla $p \in (\mu \circ \tau)[\xi]$, obraz $\tau[\xi]$ pokrywa p , czyli $\tau[\xi] \mid p$. Rozwijając definicję pokrywania musimy wskazać w $\tau[\xi]$ warunek słabszy od p lub skończony antyłańcuch maksymalny poniżej p . Z definicji części wspólnej mamy $p \in \mu[\xi] \bowtie \tau[\xi]$. Jeśli $p \in \mu[\xi]$, to wprost z definicji splotu musi istnieć $q \in \tau[\xi]$ takie, że $p \leq q$, więc $\tau[\xi] \mid p$. Jeśli $p \in \tau[\xi]$, to w oczywisty sposób jest $\tau[\xi] \mid p$. \square

Następny lemat ma przede wszystkim znaczenie dla przypadku $\alpha \subseteq \delta$, choć jest prawdziwy w całej ogólności.

LEMAT 5.4.4. Jeśli $\alpha, \delta \in \mathfrak{M}\mathfrak{F}^1$, to $\delta \asymp \alpha \subseteq \delta$.

Dowód. Niech $\tau = \delta \asymp \alpha$. Zgodnie z wnioskiem 5.2.6 trzeba pokazać, że

$$\bigvee_{\check{\eta} \in \text{dom}(\tau)} \bigvee_{q \in \tau[\check{\eta}]} \delta[\check{\eta}] \mid q .$$

Wprost z definicji różnicy 5.3.7 i faktu 5.3.10 mamy, że

$$\begin{aligned} \text{dom}(\tau) &= \text{dom}(\delta \asymp \alpha) , \\ &= \text{dom}(\delta \upharpoonright_{\text{dom}(\delta) \setminus \text{dom}(\alpha)}) \cup \text{dom}(\tilde{\alpha}^\delta) , \\ &\subseteq (\text{dom}(\delta) \setminus \text{dom}(\alpha)) \cup (\text{dom}(\delta) \cap \text{dom}(\alpha)) , \\ &= \text{dom}(\delta) . \end{aligned}$$

Jeśli $\check{\eta} \in \text{dom}(\delta) \setminus \text{dom}(\alpha)$ i $q \in \tau[\check{\eta}]$, to w oczywisty sposób $\delta[\check{\eta}] \mid q$, bo w tym przypadku $q \in \delta[\check{\eta}]$. Niech więc $\check{\eta} \in \text{dom}(\delta) \cap \text{dom}(\alpha)$ i $q \in \tau[\check{\eta}]$, czyli $q \in \tilde{\alpha}^\delta[\check{\eta}]$. Zatem $q \in \delta[\check{\eta}] \cap \alpha[\check{\eta}]^\perp$ lub $q \in \max(\alpha[\check{\eta}]^\perp) \times \delta[\check{\eta}]$. W pierwszym przypadku $q \in \delta[\check{\eta}]$, więc $\delta[\check{\eta}]$ pokrywa q . W drugim przypadku istnieje $p \in \delta[\check{\eta}]$ takie, że $p \geq q$, a stąd $\delta[\check{\eta}] \mid q$. \square

5.5 Algebra Boole'a

Pokażemy, że zdefiniowane w rozdziale 5.3 operacje algebraiczne dla metazbiorów spełniają aksjomaty algebry Boole'a.

5.5.1 Właściwości splotu

Najpierw udowodnimy kilka technicznych lematów. Po pierwsze zauważmy, że splot jest przemienny.

FAKT 5.5.1. Dla $P, Q \subset \mathbb{T}$ jest $P \bowtie Q = Q \bowtie P$.

Oczywisty dowód wynika bezpośrednio z definicji. Półsplot i splot są również rozdzielne względem sumy.

LEMAT 5.5.2. Dla $P, Q, R \subset \mathbb{T}$ zachodzą następujące równości:

$$P \times (Q \cup R) = P \times Q \cup P \times R , \quad (5.5.1)$$

$$P \bowtie (Q \cup R) = P \bowtie Q \cup P \bowtie R , \quad (5.5.2)$$

$$P \bowtie (Q \cup R) = P \bowtie Q \cup P \bowtie R . \quad (5.5.3)$$

Dowód. Na wstępie zauważmy, że jeśli $P = \emptyset$, to zarówno $P \bowtie (Q \cup R) = \emptyset$, jak i $P \bowtie Q = P \bowtie R = \emptyset$. Jeśli $Q = \emptyset$, to $P \bowtie Q = \emptyset$ i równość jest spełniona. Podobnie dla $R = \emptyset$. To samo dotyczy pozostałych operatorów: \times i \bowtie . Zatem możemy zakładać, że wszystkie zbiory P, Q, R są niepuste.

Najpierw udowodnimy równość (5.5.1). Weźmy $s \in P \times (Q \cup R)$ i założmy, że $s \in P$. Z definicji istnieje $t \geq s$ takie, że $t \in Q \cup R$. Jeśli $t \in Q$, to $s \in P \times Q$, a jeśli $t \in R$, to $s \in P \times R$. Stąd $P \times (Q \cup R) \subset P \times Q \cup P \times R$. Z drugiej strony, jeśli $t \in P \times Q$, to tym bardziej $P \times (Q \cup R)$, tak samo jeśli $P \times R$. Stąd mamy $P \times Q \cup P \times R \subset P \times (Q \cup R)$.

Analogicznie dowodzimy drugą równość (5.5.2). Aby udowodnić (5.5.3) przedstawmy splot — korzystając z definicji 5.3.3 — przy pomocy półsplotów, przyjmując jednocześnie następujące oznaczenia:

$$\begin{aligned} \overbrace{P \bowtie (Q \cup R)}^L &= \overbrace{P \times (Q \cup R)}^{L_L} \cup \overbrace{P \bowtie (Q \cup R)}^{L_P}, \\ \overbrace{P \times Q}^{P_{LL}} \cup \overbrace{P \times R}^{P_{LP}} &\cup \overbrace{P \times Q}^{P_{PL}} \cup \overbrace{P \times R}^{P_{PP}} = \overbrace{P \times Q}^{P_L} \cup \overbrace{P \times R}^{P_P}. \end{aligned}$$

Trzeba pokazać, że $L = P_L \cup P_P$, czyli:

$$L_L \cup L_P = P_{LL} \cup P_{LP} \cup P_{PL} \cup P_{PP}.$$

Równość tę otrzymamy dodając stronami równości (5.5.1) i (5.5.2). \square

Splot jest łączny, co za chwilę udowodnimy. Do dowodu będą potrzebne poniższe właściwości splotu i półsplotu.

FAKT 5.5.3. Dla dowolnych $P, Q, R \subset \mathbb{T}$ jest $(P \times Q) \times R = (P \times R) \times Q$.

Dowód. Jeśli którykolwiek ze zbiorów P, Q, R jest pusty, to lewa i prawa strona każdej równości też jest zbiorem pustym. Możemy zatem zakładać, że zbiory te są niepuste.

Jeśli $p \in (P \times Q) \times R$, to $p \in P$ oraz $\exists_{q \in Q} q \geq p$ i $\exists_{r \in R} r \geq p$. Możliwe są dwa przypadki: $p \leq q \leq r$ i $p \leq r \leq q$. To, że $p \in (P \times R) \times Q$ również oznacza, że $p \in P$ oraz $\exists_{r \in R} r \geq p$ i $\exists_{q \in Q} q \geq p$. Zatem lewa i prawa strona równości reprezentują ten sam podzbiór zbioru P . \square

FAKT 5.5.4. Dla dowolnych $P, Q, R \subset \mathbb{T}$ jest $(P \times Q) \times R = P \times (Q \bowtie R)$.

Dowód. Podobnie jak poprzednio możemy zakładać, że P, Q, R są niepuste.

Jeśli $p \in (P \times Q) \times R$, to $p \in P$ oraz $\exists_{q \in Q} q \geq p$ i $\exists_{r \in R} r \geq p$, i możliwe są dwa przypadki: $p \leq q \leq r$ i $p \leq r \leq q$. W pierwszym przypadku mamy $p \in P \times (Q \times R) \subset P \times (Q \bowtie R)$ a w drugim $p \in P \times (R \times Q) \subset P \times (Q \bowtie R)$.

Jeśli $p \in P \times (Q \bowtie R)$, to $p \in P$, i istnieje $s \in Q \bowtie R$ takie, że $p \leq s$. Z kolei to, że $s \in Q \bowtie R$ oznacza, że $s = q \in Q$ lub $s = r \in R$. W pierwszym przypadku istnieje $r \in R$ takie, że $s = q \leq r$, więc mamy $p \leq q \leq r$. W drugim przypadku $\exists_{q \in Q} s = r \leq q$ i zachodzi $p \leq r \leq q$. Z pierwszej części dowodu wynika, że w obu przypadkach $p \in (P \times Q) \times R$. \square

LEMAT 5.5.5. Splot jest łączny, tzn. dla dowolnych $P, Q, R \subset \mathbb{T}$ zachodzi:

$$(P \bowtie Q) \bowtie R = P \bowtie (Q \bowtie R).$$

Dowód. Jeśli którykolwiek ze zbiorów P, Q, R jest pusty, to lewa i prawa strona każdej równości też jest zbiorem pustym. Możemy zatem zakładać, że zbiory te są niepuste.

Przedstawmy splot w postaci sumy półsplotów, przyjmując jednocześnie poniższe oznaczenia (korzystamy tu z lematu 5.5.2):

$$\begin{aligned} \overbrace{(P \bowtie Q) \bowtie R}^L &= \overbrace{(P \times Q \cup P \times Q) \times R}^{L_L} \cup \overbrace{(P \bowtie Q) \times R}^{L_P} , \\ &= \overbrace{(P \times Q) \times R}^{L_{LL}} \cup \overbrace{(Q \times P) \times R}^{L_{LP}} \cup \overbrace{R \times (P \bowtie Q)}^{L_{PP}} , \\ \overbrace{P \bowtie (Q \bowtie R)}^P &= \overbrace{P \times (Q \bowtie R)}^{P_L} \cup \overbrace{P \times (Q \times R \cup Q \times R)}^{P_P} , \\ &= \overbrace{P \times (Q \bowtie R)}^{P_L} \cup \overbrace{(Q \times R) \times P}^{P_{PL}} \cup \overbrace{(R \times Q) \times P}^{P_{PP}} . \end{aligned}$$

Na mocy faktu 5.5.4 mamy $L_{LL} = P_L$, fakt 5.5.3 daje $L_{LP} = P_{PL}$, a połączenie obu tych faktów daje $L_{PP} = P_{PP}$. \square

5.5.2 Ciało metazbiorów

Przeniesiemy na grunt metazbiorów definicję ciała zbiorów.

DEFINICJA 5.5.1. Niech $\delta \in \mathfrak{M}\mathfrak{F}^1$ będzie niepustym metazbiorem, a $\mathcal{D} \subset \mathfrak{M}\mathfrak{F}^1$ będzie niepustą rodziną metapodzbiorów δ (tzn. $\lambda \in \mathcal{D} \rightarrow \lambda \subseteq \delta$). Rodzinę \mathcal{D} nazywamy *ciałem metazbiorów na δ* , jeśli spełnione są następujące aksjomaty:

$$\lambda \in \mathcal{D} \rightarrow \delta \prec \lambda \in \mathcal{D} , \quad (5.5.4)$$

$$\lambda \in \mathcal{D} \wedge \rho \in \mathcal{D} \rightarrow \lambda \cup \rho \in \mathcal{D} , \quad (5.5.5)$$

$$\lambda \in \mathcal{D} \wedge \rho \in \mathcal{D} \rightarrow \lambda \cap \rho \in \mathcal{D} . \quad (5.5.6)$$

Definicja zwykłego ciała zbiorów zwykle zakłada tylko dwa z powyższych aksjomatów: 5.5.4 i 5.5.5 lub 5.5.4 i 5.5.6, gdyż trzeci z nich wynika na mocy praw De Morgana. W świecie metazbiorów musimy przyjąć wszystkie trzy aksjomaty. Wynika to z faktu, że prawa De Morgana są spełnione z metarównością, a nie ze zwykłą, ostrą równością. Skutkiem tego, część wspólna dopełnień dwóch metazbiorów nie jest dopełnieniem ich sumy, chociaż jest jej równa w sensie relacji \approx . To z kolei oznacza, że jeśli dla ciała metazbiorów założymy tylko aksjomaty 5.5.4 i 5.5.6, to suma dwóch metazbiorów z takiego ciała \mathcal{D} może nie należeć do niego, choć \mathcal{D} będzie zawierać metazbiór równy (w sensie \approx) takiej sumie. Pokazuje to następujący przykład.

PRZYKŁAD 5.5.1: Niech $\check{\mu} \in \mathfrak{M}\mathfrak{F}^c$ będzie dowolnym dziedzicznie skończonym metazbiorem kanonicznym. Rozważmy następujące metazbiory:

$$\delta = \{ \langle \check{\mu}, \mathbb{1} \rangle \} ,$$

$$\xi = \{ \langle \check{\mu}, [0] \rangle \} ,$$

$$\eta = \{ \langle \check{\mu}, [1] \rangle \} .$$

Oczywiście $\delta, \xi, \eta \in \mathfrak{M}\mathfrak{F}^1$. Oprócz tego mamy: $\xi \subseteq \delta$ i $\eta \subseteq \delta$, jak również $\emptyset \subseteq \delta$. Przyjmijmy zatem $\mathcal{D} = \{ \emptyset, \xi, \eta, \delta \}$. Pokażemy, że \mathcal{D} jest zamknięte na operacje

\cap i \simeq , ale nie jest zamknięta na \cup . Najpierw zauważmy, że dla $\alpha \in \mathcal{D}$:

$$\begin{aligned}\emptyset \cap \alpha &= \emptyset, \\ \delta \cap \alpha &= \alpha.\end{aligned}$$

Pierwsza równość jest oczywista, druga wynika z faktu, że $\delta[\check{\mu}] \bowtie \alpha[\check{\mu}] = \alpha[\check{\mu}]$ (bo $\delta[\check{\mu}] = \mathbb{1}$). Z kolei $\xi \cap \eta = \emptyset$, gdyż $\xi[\check{\mu}] \bowtie \eta[\check{\mu}] = \emptyset$ ($\xi[\check{\mu}]$ i $\eta[\check{\mu}]$ składają się z nieporównywalnych warunków). Pokazaliśmy w ten sposób, że \mathcal{D} jest zamknięta na operację \cap , tzn. $\alpha, \beta \in \mathcal{D} \rightarrow \alpha \cap \beta \in \mathcal{D}$. Dalej zauważmy, że dla $\alpha \in \mathcal{D}$:

$$\begin{aligned}\emptyset \simeq \alpha &= \emptyset, \\ \alpha \simeq \emptyset &= \alpha, \\ \alpha \simeq \alpha &= \emptyset.\end{aligned}$$

Powyższe równości są oczywiste. Do kompletu pozostało udowodnić, że:

$$\begin{aligned}\delta \simeq \xi &= \eta, & \delta \simeq \eta &= \xi, \\ \xi \simeq \eta &= \xi, & \eta \simeq \xi &= \eta.\end{aligned}$$

Pierwsze dwie równości wynikają z tego, że $\tilde{\xi}^\delta = \eta$ i $\tilde{\eta}^\delta = \xi$. Rzeczywiście:

$$\begin{aligned}\delta \simeq \xi &= \delta \upharpoonright_{\text{dom}(\delta) \setminus \text{dom}(\xi)} \cup \tilde{\xi}^\delta, \\ &= \delta \upharpoonright_{\emptyset} \cup \tilde{\xi}^\delta, \\ &= \tilde{\xi}^\delta, \\ &= \{ \langle \check{\mu}, p \rangle : p \in \delta[\check{\mu}] \cap \xi[\check{\mu}]^\perp \cup \max(\xi[\check{\mu}]^\perp) \times \delta[\check{\mu}] \}, \\ &= \{ \langle \check{\mu}, p \rangle : p \in \{ \mathbb{1} \} \cap \{ [0] \}^\perp \cup \{ [1] \} \times \{ \mathbb{1} \} \}, \\ &= \{ \langle \check{\mu}, p \rangle : p \in \emptyset \cup \{ [1] \} \}, \\ &= \{ \langle \check{\mu}, [1] \rangle \}, \\ &= \eta.\end{aligned}$$

Podobnie w drugim przypadku. Pozostałe dwie równości wynikają z faktu, że $\tilde{\eta}^\xi = \xi$ i $\tilde{\xi}^\eta = \eta$. Dowód podobny jak poprzednio, na przykład:

$$\begin{aligned}\tilde{\eta}^\xi &= \{ \langle \check{\mu}, p \rangle : p \in \xi[\check{\mu}] \cap \eta[\check{\mu}]^\perp \cup \max(\eta[\check{\mu}]^\perp) \times \xi[\check{\mu}] \}, \\ &= \{ \langle \check{\mu}, p \rangle : p \in \{ [0] \} \cap \{ [0] \}^\perp \cup \{ [0] \} \times \{ [0] \} \}, \\ &= \{ \langle \check{\mu}, p \rangle : p \in \emptyset \cup \{ [0] \} \}, \\ &= \{ \langle \check{\mu}, [0] \rangle \}, \\ &= \xi.\end{aligned}$$

Pokazaliśmy więc, że \mathcal{D} jest zamknięta na \simeq , czyli $\alpha, \beta \in \mathcal{D} \rightarrow \alpha \simeq \beta \in \mathcal{D}$. Teraz pokażemy, że \mathcal{D} nie jest zamknięta na operację \cup . Rzeczywiście, mimo iż $\xi, \eta \in \mathcal{D}$, to jednak $\xi \cup \eta \notin \mathcal{D}$, gdyż $\xi \cup \eta = \xi \cup \eta = \{ \langle \check{\mu}, [0] \rangle, \langle \check{\mu}, [1] \rangle \}$. Takiego zbioru nie ma w \mathcal{D} .

Zachodzi jednak $\xi \cup \eta \approx \delta$. Aby to uzasadnić, należy wykazać równoważność zbiorów $(\xi \cup \eta)[\check{\mu}]$ i $\delta[\check{\mu}]$, czyli $(\xi \cup \eta)[\check{\mu}] \parallel \delta[\check{\mu}]$ (por. def. 5.1.5). Wziąwszy

pod uwagę, że $(\xi \cup \eta)[\check{\mu}] = \{[0], [1]\}$, oraz $\delta[\check{\mu}] = \{\mathbb{1}\}$, łatwo widzieć, że $\{[0], [1]\} \mid \mathbb{1}$ (bo $\{[0], [1]\}$ rozpina $\mathbb{1}$, czyli zawiera skończony antylańcuch maksymalny poniżej $\mathbb{1}$), $\{\mathbb{1}\} \mid [0]$ oraz $\{\mathbb{1}\} \mid [1]$ (bo $\{\mathbb{1}\}$ zawiera warunek słabszy od $[0]$ i $[1]$), co razem daje $\{[0], [1]\} \parallel \{\mathbb{1}\}$.

Podsumowując: widzimy na podstawie powyższego przykładu, że wymaganie spełnienia wszystkich trzech aksjomatów w definicji 5.5.1 jest konieczne.

Podobnie jak ciało ostrych zbiorów, ciało metazbiorów na δ zawiera metazbiór pusty, jak również samo δ . Aby to udowodnić, wykażemy najpierw inną właściwość.

LEMAT 5.5.6. Jeśli $\alpha, \delta \in \mathfrak{M}\mathfrak{F}^\perp$, to $\alpha \circ (\delta \prec \alpha) = \emptyset$.

Dowód. Pokażemy, że dla dowolnego $\check{\mu} \in \mathfrak{M}\mathfrak{F}^c$ jest $\alpha[\check{\mu}] \bowtie (\delta \prec \alpha)[\check{\mu}] = \emptyset$, co wystarczy do udowodnienia tezy, bo $(\alpha \circ (\delta \prec \alpha))[\check{\mu}] = \alpha[\check{\mu}] \bowtie (\delta \prec \alpha)[\check{\mu}]$.

Najpierw zauważmy, że fakt 5.3.4 implikuje $\text{dom}(\alpha \circ (\delta \prec \alpha)) \subset \text{dom}(\alpha)$. W takim razie, dla $\check{\mu} \notin \text{dom}(\alpha)$ zachodzi $\alpha[\check{\mu}] \bowtie (\delta \prec \alpha)[\check{\mu}] = \emptyset$. Niech więc $\check{\mu} \in \text{dom}(\alpha)$. Zauważmy, że $(\delta \prec \alpha) \upharpoonright_{\text{dom}(\alpha)} = \tilde{\alpha}^\delta$, w związku z czym wystarczy pokazać, że $\alpha[\check{\mu}] \bowtie \tilde{\alpha}^\delta[\check{\mu}] = \emptyset$ czyli, że $\alpha[\check{\mu}] \times \tilde{\alpha}^\delta[\check{\mu}] = \emptyset$ i $\alpha[\check{\mu}] \times \tilde{\alpha}^\delta[\check{\mu}] = \emptyset$. Z definicji brzegu 5.3.6 mamy:

$$\tilde{\alpha}^\delta[\check{\mu}] = \delta[\check{\mu}] \cap \alpha[\check{\mu}]^\perp \cup \max(\alpha[\check{\mu}]^\perp) \times \delta[\check{\mu}] \subset \alpha[\check{\mu}]^\perp.$$

Ponadto $\alpha[\check{\mu}] \times \alpha[\check{\mu}]^\perp = \emptyset$, bo nad warunkiem z $\alpha[\check{\mu}]$ nie może występować żaden element ze zbioru warunków nieporównywalnych z $\alpha[\check{\mu}]$, czyli z $\alpha[\check{\mu}]^\perp$. Stąd otrzymujemy $\alpha[\check{\mu}] \times \tilde{\alpha}^\delta[\check{\mu}] \subset \alpha[\check{\mu}] \times \alpha[\check{\mu}]^\perp = \emptyset$.

Podobnie $\alpha[\check{\mu}] \times \alpha[\check{\mu}]^\perp = \emptyset$, bo jeśli $p \in \alpha[\check{\mu}]^\perp$, to p jest z definicji nieporównywalny z żadnym warunkiem z $\alpha[\check{\mu}]$, więc nie może mieć nad sobą elementu z $\alpha[\check{\mu}]$. Stąd mamy $\alpha[\check{\mu}] \times \tilde{\alpha}^\delta[\check{\mu}] \subset \alpha[\check{\mu}] \times \alpha[\check{\mu}]^\perp = \emptyset$. \square

LEMAT 5.5.7. Jeśli \mathcal{D} jest ciałem metazbiorów na δ , to $\emptyset \in \mathcal{D}$ i $\delta \in \mathcal{D}$.

Dowód. Ciało metazbiorów jest z definicji niepuste. Niech więc $\xi \in \mathcal{D}$. Wtedy również że również $\delta \prec \xi \in \mathcal{D}$. Z lematu 5.5.6 wiemy ($\xi \subseteq \delta$, bo $\xi \in \mathcal{D}$), że $\xi \circ (\delta \prec \xi) = \emptyset$. Z zamkniętości \mathcal{D} na \circ mamy, że $\emptyset \in \mathcal{D}$. Stąd również $\delta \prec \emptyset = \delta \in \mathcal{D}$. \square

Jak wiadomo, rodzina podzbiorów danego zbioru jest ciałem zbiorów. Podobnie jest w przypadku metazbiorów.

TWIERDZENIE 5.5.8. Niech $\delta \in \mathfrak{M}\mathfrak{F}^\perp$. Rodzina $\mathcal{D} = \{\lambda \in \mathfrak{M}\mathfrak{F}^\perp : \lambda \subseteq \delta\}$ złożona ze wszystkich metapodzbiorów klasy $\mathfrak{M}\mathfrak{F}^\perp$ w δ jest ciałem metazbiorów.

Dowód. Należy pokazać, że spełnione są wszystkie aksjomaty z definicji 5.5.1. Weźmy $\lambda \in \mathcal{D}$. Z lematu 5.4.4 wynika, że $\delta \prec \lambda \subseteq \delta$, co razem z faktem 5.3.13 implikuje $(\delta \prec \lambda) \in \mathcal{D}$. To dowodzi pierwszego aksjomatu (5.5.4). Dla $\lambda, \rho \in \mathcal{D}$ fakt 5.3.1 i lemat 5.4.1 dowodzą aksjomatu (5.5.5). Fakt 5.3.5 oraz lematy 5.4.3 i 4.2.9 implikują (5.5.6). \square

5.5.3 Twierdzenie podstawowe

Dla ustalenia uwagi przytoczymy teraz definicję algebry Boole'a.

DEFINICJA 5.5.2. Algebra Boole'a to szóstka $\mathbb{B} = (\mathcal{B}, +, \cdot, \sim, 0, 1)$, w której $+$ i \cdot są działaniami dwuargumentowymi, \sim jest operacją jednoargumentową, 0 i 1 są wyróżnionymi, różnymi elementami zbioru \mathcal{B} , taka, że następujące warunki są spełnione dla wszystkich $a, b, c \in \mathcal{B}$:

$$\begin{array}{lll} a + (b + c) = (a + b) + c, & a \cdot (b \cdot c) = (a \cdot b) \cdot c, & (\text{łącność}) \\ a + b = b + a, & a \cdot b = b \cdot a, & (\text{przemienność}) \\ a + (a \cdot b) = a, & a \cdot (a + b) = a, & (\text{absorpcja}) \\ a + (b \cdot c) = (a + b) \cdot (a + c), & a \cdot (b + c) = (a \cdot b) + (a \cdot c), & (\text{rozdzielność}) \\ a + \sim a = 1, & a \cdot \sim a = 0. & (\text{odwrotność}) \end{array}$$

Pokażemy, że ciało metazbiorów na δ , dla dziedzicznie skończonego metazbioru pierwszego poziomu δ , jest algebrą Boole'a. Operacjami dwuargumentowymi reprezentowanymi przez symbole $+$ i \cdot będą w tym przypadku \cup i \cap . Za operację jednoargumentową $\sim \lambda$ przyjmiemy różnicę $\delta \setminus \lambda$. Wyróżnionymi elementami 0 i 1 będą \emptyset i δ . Symbol równości $=$ w aksjomatach będzie odnosił się do równości metazbiorów \approx .

Jak zobaczymy dalej, równości w niektórych aksjomatach będą ostrymi równościami mnogościowymi, a nie tylko równościami dla metazbiorów. Takie ostre równości będą wyrażać tożsamość metazbiorów występujących po lewej i prawej stronie. W przypadku metarówności lewa i prawa strona nie muszą być identycznymi zbiorami, są tylko równe jako metazbiory. Tam, gdzie będzie zachodziła ostra równość, zostanie to podkreślone.

TWIERDZENIE 5.5.9. Niech $\delta \in \mathfrak{M}\mathfrak{S}^1$, a \mathcal{D} będzie ciałem metazbiorów na δ . Dla $\alpha, \beta, \gamma \in \mathcal{D}$ spełnione są następujące równości:

$$\alpha \cup (\beta \cap \gamma) \approx (\alpha \cup \beta) \cap \gamma, \quad (5.5.7)$$

$$\alpha \cap (\beta \cup \gamma) \approx (\alpha \cap \beta) \cup \gamma, \quad (5.5.8)$$

$$\alpha \cup \beta \approx \beta \cup \alpha, \quad (5.5.9)$$

$$\alpha \cap \beta \approx \beta \cap \alpha, \quad (5.5.10)$$

$$\alpha \cup (\alpha \cap \beta) \approx \alpha, \quad (5.5.11)$$

$$\alpha \cap (\alpha \cup \beta) \approx \alpha, \quad (5.5.12)$$

$$\alpha \cup (\beta \cap \gamma) \approx (\alpha \cup \beta) \cap (\alpha \cup \gamma), \quad (5.5.13)$$

$$\alpha \cap (\beta \cup \gamma) \approx (\alpha \cap \beta) \cup (\alpha \cap \gamma), \quad (5.5.14)$$

$$\alpha \cup (\delta \setminus \alpha) \approx \delta, \quad (5.5.15)$$

$$\alpha \cap (\delta \setminus \alpha) \approx \emptyset. \quad (5.5.16)$$

Zatem \mathcal{D} jest algebrą Boole'a.

Dowód twierdzenia rozbijemy na kilka lematów.

LEMAT 5.5.10. Jeśli $\alpha, \beta, \gamma \in \mathfrak{M}\mathfrak{S}^1$, to $\alpha \cap (\beta \cap \gamma) = (\alpha \cap \beta) \cap \gamma$.

Dowód. Przyjmijmy oznaczenia: $\eta = \beta \cap \gamma$, $\xi = \alpha \cap \beta$, $\lambda = \alpha \cap \eta$ i $\rho = \xi \cap \gamma$. Pokażemy, że dla dowolnego $\check{\mu} \in \mathfrak{M}^c$ jest $\lambda[\check{\mu}] = \rho[\check{\mu}]$. Z definicji części wspól-

nej 5.3.4 i lematu 5.5.5 otrzymujemy:

$$\begin{aligned}
\lambda[\check{\mu}] &= \alpha[\check{\mu}] \bowtie \eta[\check{\mu}] , \\
&= \alpha[\check{\mu}] \bowtie (\beta[\check{\mu}] \bowtie \gamma[\check{\mu}]) , \\
&= (\alpha[\check{\mu}] \bowtie \beta[\check{\mu}]) \bowtie \gamma[\check{\mu}] , \\
&= \xi[\check{\mu}] \bowtie \gamma[\check{\mu}] , \\
&= \rho[\check{\mu}] .
\end{aligned}$$

Z tego wynika, że $\lambda = \rho$, bo metazbiór jest jednoznacznie wyznaczony przez swoje elementy potencjalne i ich obrazy, a w przypadku gdy $\lambda[\check{\mu}] = \emptyset = \rho[\check{\mu}]$ metazbiór $\check{\mu}$ nie jest elementem potencjalnym λ ani ρ . \square

Zwróćmy uwagę, że udowodniliśmy ostrą równość $=$, a nie tylko równość dla metazbiorów \approx wymaganą w twierdzeniu 5.5.9. W związku z powyższym możemy pomijając nawiasy używać notacji:

$$\alpha \circ \beta \circ \gamma = \alpha \circ (\beta \circ \gamma) = (\alpha \circ \beta) \circ \gamma . \quad (5.5.17)$$

Następna równość również jest ostra.

LEMAT 5.5.11. Jeśli $\alpha, \beta, \gamma \in \mathfrak{M}\mathfrak{F}^1$, to $\alpha \circ (\beta \cup \gamma) = (\alpha \circ \beta) \cup (\alpha \circ \gamma)$.

Dowód. Przyjmijmy oznaczenia: $\lambda_1 = \beta \cup \gamma$, $\rho_1 = \alpha \circ \beta$, $\rho_2 = \alpha \circ \gamma$, $\lambda = \alpha \circ \lambda_1$ i $\rho = \rho_1 \cup \rho_2$. Pokażemy, że dla dowolnego $\check{\mu} \in \mathfrak{M}^c$ jest $\lambda[\check{\mu}] = \rho[\check{\mu}]$.

Najpierw zauważmy, że dla dowolnych metazbiorów τ, σ, μ jest:

$$(\tau \cup \sigma)[\mu] = (\tau \cup \sigma)[\mu] = \tau[\mu] \cup \sigma[\mu] .$$

Z tego i z lematu 5.5.2 otrzymujemy dla dowolnego $\check{\mu} \in \mathfrak{M}^c$:

$$\begin{aligned}
\lambda[\check{\mu}] &= \alpha[\check{\mu}] \bowtie \lambda_1[\check{\mu}] , \\
&= \alpha[\check{\mu}] \bowtie (\beta[\check{\mu}] \cup \gamma[\check{\mu}]) , \\
&= (\alpha[\check{\mu}] \bowtie \beta[\check{\mu}]) \cup (\alpha[\check{\mu}] \bowtie \gamma[\check{\mu}]) , \\
&= \rho_1[\check{\mu}] \cup \rho_2[\check{\mu}] , \\
&= \rho[\check{\mu}] .
\end{aligned}$$

To kończy dowód, bo metazbiór jest jednoznacznie wyznaczony przez swoje elementy potencjalne i ich obrazy. \square

W przypadku aksjomatu (5.5.11) z twierdzenia 5.5.9 czysta równość może nie zachodzić, gdyż $\alpha \cup (\alpha \circ \beta) = \alpha$ wymagałoby $\alpha \circ \beta \subset \alpha$. Ponieważ dla dowolnego $\mu \in \text{dom}(\alpha) \cap \text{dom}(\beta)$ jest $(\alpha \circ \beta)[\mu] = \alpha[\mu] \bowtie \beta[\mu] \supset \alpha[\mu] \times \beta[\mu]$, a ostatni zbiór może zawierać elementy spoza $\alpha[\mu]$, więc również spoza $\text{ran}(\alpha)$, to może się zdarzyć, że $(\alpha \circ \beta) \setminus \alpha \neq \emptyset$. Mamy jednak równość przybliżoną.

LEMAT 5.5.12. Jeśli $\alpha, \beta \in \mathfrak{M}\mathfrak{F}^1$, to $\alpha \cup (\alpha \circ \beta) \approx \alpha$.

Dowód. Przyjmijmy oznaczenie $\tau = \alpha \cup (\alpha \circ \beta)$. Jest $\text{dom}(\tau) = \text{dom}(\alpha)$, gdyż $\text{dom}(\alpha \circ \beta) \subset \text{dom}(\alpha)$ (fakty 5.3.2, 5.3.4). Zatem zgodnie z wnioskiem 5.2.9 trzeba pokazać, że dla $\mu \in \text{dom}(\alpha)$ zachodzi $\tau[\mu] \parallel \alpha[\mu]$, czyli, że zbiory te są równoważne. A więc dla $p \in \alpha[\mu]$ mamy pokazać $\tau[\mu] \upharpoonright p$, czyli, że $\tau[\mu]$ zawiera skończony antyłańcuch maksymalny poniżej p , lub $\tau[\mu]$ zawiera warunek słabszy

niż p . Podobnie dla $q \in \tau[\mu]$ ma być $\alpha[\mu] \mid q$. Pierwsza część jest oczywista, bo $\alpha[\mu] \subset \tau[\mu]$, więc jeśli $p \in \alpha[\mu]$, to również $p \in \tau[\mu]$. Weźmy w takim razie $q \in \tau[\mu]$. Jeśli $q \in \alpha[\mu]$, to oczywiście $\alpha[\mu] \mid q$. W przeciwnym przypadku $q \in (\alpha \cap \beta)[\mu]$ i wtedy $q \in \beta[\mu] \times \alpha[\mu]$, bo $q \notin \beta[\mu] \times \alpha[\mu]$. Zatem istnieje $r \in \alpha[\mu]$ takie, że $q \leq r$. Stąd $\alpha[\mu] \mid q$, i ostatecznie $\tau[\mu] \parallel \alpha[\mu]$, co daje $\tau \approx \alpha$. \square

LEMAT 5.5.13. Jeśli $\alpha, \delta \in \mathfrak{M}\mathfrak{F}^1$, to $\alpha \cup (\delta \asymp \alpha) \approx \delta$.

Dowód. Przyjmijmy oznaczenie: $\lambda = \alpha \cup (\delta \asymp \alpha)$. Po pierwsze zauważmy, że $\text{dom}(\lambda) = \text{dom}(\delta)$. Istotnie, ponieważ $\alpha \subseteq \delta$, to z definicji sumy \cup i z faktu 5.3.12 mamy:

$$\text{dom}(\lambda) = \text{dom}(\alpha) \cup \text{dom}(\delta \asymp \alpha) \subset \text{dom}(\delta) .$$

Z drugiej strony, z definicji różnicy \asymp mamy:

$$\begin{aligned} \text{dom}(\delta) &= (\text{dom}(\delta) \setminus \text{dom}(\alpha)) \cup \text{dom}(\alpha) , \\ &\subset \text{dom}(\delta \asymp \alpha) \cup \text{dom}(\alpha) , \\ &= \text{dom}(\lambda) . \end{aligned}$$

Niech $\check{\mu} \in \text{dom}(\lambda)$. Żeby udowodnić przybliżoną równość $\lambda \approx \delta$, musimy wykazać $\lambda[\check{\mu}] \parallel \delta[\check{\mu}]$, czyli równoważność obrazów $\lambda[\check{\mu}]$ i $\delta[\check{\mu}]$. Z definicji równoważności 5.1.5 oznacza to, że dla $p \in \lambda[\check{\mu}]$ ma być $\delta[\check{\mu}] \mid p$ ($\delta[\check{\mu}]$ pokrywa p), oraz dla $q \in \delta[\check{\mu}]$ ma być $\lambda[\check{\mu}] \mid q$. Zgodnie z definicją pokrywania 5.1.4 musimy pokazać, że $\delta[\check{\mu}]$ zawiera skończony antylańcuch maksymalny poniżej p lub jakiś warunek powyżej p , i podobnie dla zbioru $\lambda[\check{\mu}]$ i warunku q .

Niech $p \in \lambda[\check{\mu}]$. Zauważmy, że $\lambda[\check{\mu}] = \alpha[\check{\mu}] \cup (\delta \asymp \alpha)[\check{\mu}]$. Jeśli $p \in \alpha[\check{\mu}]$, to $\delta[\check{\mu}] \mid p$, bo $\alpha \subseteq \delta$ (wniosek 5.2.6). W przeciwnym przypadku $p \in (\delta \asymp \alpha)[\check{\mu}]$. Wtedy jeśli $\check{\mu} \in \text{dom}(\delta) \setminus \text{dom}(\alpha)$, to $p \in \delta[\check{\mu}]$ i w oczywisty sposób $\delta[\check{\mu}] \mid p$. Natomiast jeśli $p \in \text{dom}(\delta) \cap \text{dom}(\alpha)$, to $p \in \tilde{\alpha}^\delta[\check{\mu}]$ i wtedy, zgodnie z definicją 5.3.6, $p \in \delta[\check{\mu}]$ lub $\exists_{q \geq p} q \in \delta[\check{\mu}]$. W obu przypadkach $\delta[\check{\mu}] \mid p$.

Niech teraz $q \in \delta[\check{\mu}]$. Pokażemy, że $\lambda[\check{\mu}] \mid q$. Z definicji różnicy otrzymujemy:

$$\begin{aligned} \lambda[\check{\mu}] &= \alpha[\check{\mu}] \cup (\delta \asymp \alpha)[\check{\mu}] , \\ &= \alpha[\check{\mu}] \cup \delta \upharpoonright_{\text{dom}(\delta) \setminus \text{dom}(\alpha)}[\check{\mu}] \cup \tilde{\alpha}^\delta[\check{\mu}] . \end{aligned}$$

Jeśli $\check{\mu} \notin \text{dom}(\alpha)$, to $\alpha[\check{\mu}] = \tilde{\alpha}^\delta[\check{\mu}] = \emptyset$, więc $\lambda[\check{\mu}] = \delta[\check{\mu}]$ i mamy $\lambda[\check{\mu}] \mid p$. Zakładamy zatem, że $\check{\mu} \in \text{dom}(\alpha)$. Jeśli $q \in \alpha[\check{\mu}]^\perp$, to zgodnie z definicją 5.3.6 $q \in \tilde{\alpha}^\delta[\check{\mu}]$ (bo $q \in \delta[\check{\mu}] \cap \alpha[\check{\mu}]^\perp$), i w tej sytuacji mamy $\lambda[\check{\mu}] \mid q$. Niech więc $q \in \alpha[\check{\mu}]^\top$, czyli q jest porównywalne z jakimś warunkiem z $\alpha[\check{\mu}]$. Jeśli istnieje $r \geq q$ takie, że $r \in \alpha[\check{\mu}]$, to oczywiście $\lambda[\check{\mu}] \mid q$, bo $\lambda[\check{\mu}]$ zawiera r . W przeciwnym przypadku musi istnieć $r < q$ takie, że $r \in \alpha[\check{\mu}]$. Mamy zatem warunek z $\lambda[\check{\mu}]$ poniżej q i żadnego powyżej (bo $\tilde{\alpha}^\delta[\check{\mu}] \subset \alpha[\check{\mu}]^\perp$, a $q \in \alpha[\check{\mu}]^\top$). Pokażemy, że $R = \{r \leq q: r \in \lambda[\check{\mu}]\}$ rozpina q , czyli, że zawiera skończony antylańcuch maksymalny poniżej q . To będzie implikować $\lambda[\check{\mu}] \mid q$.

Niech $S = \{s \leq q: s \in \alpha[\check{\mu}]^\perp\}$. Jeśli $S = \emptyset$, to każdy warunek silniejszy od q jest porównywalny z jakimś elementem $\alpha[\check{\mu}]$, z założenia leżącym poniżej q . Zbiór $\max(\alpha[\check{\mu}])$ rozpina q , bo zawiera antylańcuch poniżej q , który jest maksymalny (poprzednie zdanie) i skończony, gdyż $\alpha \in \mathfrak{M}\mathfrak{F}^1$. Również zbiór $R \cap \max(\alpha[\check{\mu}])$, a tym samym i R mają tę właściwość, dzięki czemu w przypadku, gdy S jest pusty mamy $\lambda[\check{\mu}] \mid q$.

Przyjmijmy więc, że $S \neq \emptyset$. Mamy: $\max(S) \subset \tilde{\alpha}^\delta[\tilde{\mu}]$, bo dla $s \in \max(S)$ jest $s \in \max(\alpha[\tilde{\mu}]^\perp) \times \delta[\tilde{\mu}]$, gdyż $s \leq q$ i $q \in \delta[\tilde{\mu}]$. Zatem $\max(S) \subset R$ i $\max(S)$ jest skończonym antylańcuchem (lemat 5.3.9). Zbiór $R \cap \max(\alpha[\tilde{\mu}])$ również jest skończonym antylańcuchem, a suma $R \cap \max(\alpha[\tilde{\mu}]) \cup \max(S)$ zawiera antylańcuch maksymalny, bo każdy element albo jest porównywalny z czymś z $\alpha[\tilde{\mu}]$, albo nie jest. Ponieważ $R \cap \max(\alpha[\tilde{\mu}]) \cup \max(S) \subset R$, to R rozpina q , a więc również pokrywa q , a stąd $\lambda[\tilde{\mu}] \mid q$, co kończy dowód lematu. \square

Dowód twierdzenia 5.5.9. Przypomnijmy, że $=$ implikuje \approx .

Aksjomaty (5.5.7), (5.5.9) są oczywiste. Aksjomat (5.5.10) wynika z faktu 5.5.1. Aksjomat (5.5.8) wynika z lematu 5.5.10. Aksjomatu (5.5.11) dowodzi lemat 5.5.12. Aksjomat (5.5.12) wynika z (5.5.11) i (5.5.14) oraz z faktu 5.4.2 mówiącego, że $\alpha \cap \alpha = \alpha$, w następujący sposób:

$$\begin{aligned} \alpha \cap (\alpha \cup \beta) &\approx (\alpha \cap \alpha) \cup (\alpha \cap \beta) , \\ &= \alpha \cup (\alpha \cap \beta) , \\ &\approx \alpha . \end{aligned}$$

Aksjomatu (5.5.14) dowodzi lemat 5.5.11. Aksjomat (5.5.13) łatwo wynika z pozostałych:

$$\begin{aligned} \alpha \cup (\beta \cap \gamma) &\approx \alpha \cup (\alpha \cap \beta) \cup (\beta \cap \gamma) , \\ &\approx [\alpha \cap (\alpha \cup \gamma)] \cup [(\beta \cap \alpha) \cup (\beta \cap \gamma)] , \\ &\approx [\alpha \cap (\alpha \cup \gamma)] \cup [\beta \cap (\alpha \cup \gamma)] , \\ &\approx (\alpha \cup \beta) \cap (\alpha \cup \gamma) . \end{aligned}$$

Aksjomatu (5.5.16) dowodzi lemat 5.5.6, natomiast aksjomatu (5.5.15) dowodzi lemat 5.5.13. To kończy dowód twierdzenia. \square

Korzystając z twierdzeń 5.5.8 i 5.5.9 otrzymujemy następujący wniosek.

WNIOSEK 5.5.14. Dla $\delta \in \mathfrak{M}\mathfrak{F}^1$ rodzina jego dziedzicznie skończonych metapodzbiorów pierwszego poziomu jest algebrą Boola.

Rozdział 6

Metazbiory a zbiory rozmyte

Metazbiory uogólniają zbiory rozmyte [34]. Oznacza to, że każdy zbiór rozmyty można przedstawić jako metazbiór. Co więcej, nie tylko zwykłe zbiory rozmyte typu 1 można reprezentować w metazbiorach, ale również zbiory rozmyte typu 2 oraz intuicjonistyczne zbiory rozmyte [2].

W tym rozdziale przedstawimy metodę reprezentacji zbiorów rozmytych typu 1 określonych na skończonej przestrzeni, których funkcje przynależności przyjmują wartości dwójkowo-wymierne. Dla zastosowań komputerowych takie ograniczenie nie jest istotne. W zamian otrzymujemy zgodność podstawowych relacji i operacji.

6.1 Reprezentacje w $\mathfrak{M}\mathfrak{F}^1$

Zbiór rozmyty A typu 1 jest określony przez swoją dziedzinę X i funkcję przynależności $\mu_A : X \mapsto [0 \dots 1]$ przyporządkowującą każdemu elementowi dziedziny stopień przynależności, będący liczbą rzeczywistą z przedziału jednostkowego. Zakładamy, że dziedzina zbioru rozmytego A jest skończona: $|X| < \aleph_0$, a jego funkcja przynależności μ_A przyjmuje wartości dwójkowo-wymierne, czyli o mianownikach będących potęgami dwójki.

Aby reprezentować zbiór rozmyty A przy pomocy metazbioru ρ_A musimy najpierw zdefiniować dziedzinę dla ρ_A . Zauważmy, że natura elementów dziedziny $\text{dom}(\rho_A)$ jest bez znaczenia, choć ważne jest to, żeby były one niezależne od interpretacji. W związku z tym jako elementy potencjalne metazbioru ρ_A wybierzemy jakies meta zbiory kanoniczne. Dokładniej, będzie to zbiór kanonicznych odpowiedników liczb porządkowych mniejszych od $m = |X|$ (def. 14). Zatem:

$$\text{dom}(\rho_A) = \text{dom}(\tilde{m}) = \left\{ \check{0}, \dots, \overline{m-1} \right\} . \quad (6.1.1)$$

Zauważmy, że wszystkie elementy dziedziny $\text{dom}(\rho_A)$ są dziedzicznie skończone.

Ustalamy również (jakies) odwzorowanie dziedzin $i_X : \text{dom}(\rho_A) \mapsto X$ wzajemnie jednoznacznie przyporządkowujące elementom potencjalnym z dziedziny ρ_A elementy zbioru X . W istocie, funkcja i_X numeruje elementy zbioru X kanonicznymi odpowiednikami liczb porządkowych.

Niech $k \in \mathbb{N}$ będzie takie, że wartości funkcji przynależności $\mu_A(x)$ dla wszystkich $x \in X$ dają się przedstawić jako ułamki o mianowniku 2^k . Takie

k istnieje, bo X jest skończony, a μ_A przyjmuje wartości dwójkowo-wymierne. Dla $x \in X$, niech $l_A^k(x)$ będzie licznikiem ułamka $\mu_A(x)$ przy mianowniku równym 2^k . Oczywiście, $0 \leq l_A^k(x) \leq 2^k$ oraz $\mu_A(x) = \frac{l_A^k(x)}{2^k}$ dla $x \in X$.

Niech $p_j^k \in \mathbb{T}_k$ dla $0 \leq j < 2^k$ oznaczają elementy k -tego poziomu drzewa \mathbb{T} uporządkowane w naturalny sposób tzn. indeksowane zgodnie z porządkiem poziomu:¹ p_j^k traktowany jako binarny zapis liczby naturalnej reprezentuje liczbę będącą jego indeksem j . Na przykład dla $k = 2$ mamy $p_0^2 = [00]$, $p_1^2 = [01]$, $p_2^2 = [10]$, $p_3^2 = [11]$.

Metazbiór ρ_A reprezentujący zbiór rozmyty A definiujemy następująco (podaję kilka równoważnych notacji):

$$\rho_A = \bigcup_{x \in X} \{i_X^{-1}(x)\} \times \{p_j^k : 0 \leq j < l_A^k(x)\} , \quad (6.1.2)$$

$$= \bigcup_{\tilde{n} \in \text{dom}(\rho_A)} \{\tilde{n}\} \times \{p_j^k : 0 \leq j < l_A^k(i_X(\tilde{n}))\} , \quad (6.1.3)$$

$$= \{ \langle \tilde{n}, p_j^k \rangle : \tilde{n} \in \text{dom}(\rho_A) \wedge 0 \leq j < l_A^k(i_X(\tilde{n})) \} , \quad (6.1.4)$$

$$= \{ \langle \tilde{n}, p_j^k \rangle : 0 \leq n < |X| \wedge 0 \leq j < \mu_A(i_X(\tilde{n})) \cdot 2^k \} . \quad (6.1.5)$$

Jak widać, każdemu elementowi potencjalnemu \tilde{n} z dziedziny ρ_A przyporządkowujemy tyle początkowych warunków z \mathbb{T}_k , ile wynosi licznik (przy mianowniku postaci 2^k) wartości funkcji przynależności $\mu_A(x)$ dla elementu x będącego odpowiednikiem w zbiorze X elementu potencjalnego \tilde{n} przy odwzorowaniu i_X (tzn. $x = i_X(\tilde{n})$). W ten sposób zapewniamy, że (por. def. obrazu 3.2.5)

$$|\rho_A[\tilde{n}]| = l_A^k(\mu_A(i_X(\tilde{n}))) \quad \text{czyli} \quad \frac{|\rho_A[\tilde{n}]|}{2^k} = \mu_A(i_X(\tilde{n})) . \quad (6.1.6)$$

Zbiór warunków $\rho_A[\tilde{n}]$ określa stopień należenia elementu potencjalnego \tilde{n} do ρ_A . Jako wartość numeryczną stopnia należenia przyjmujemy tu stosunek liczby elementów obrazu $\rho_A[\tilde{n}]$ do liczby wszystkich warunków na poziomie \mathbb{T}_k .

Zwróćmy uwagę, że dla spełnienia (6.1.6) wystarczy jedynie zapewnić, że liczba warunków w obrazie $\rho_A[\tilde{n}]$ będzie odpowiednia, natomiast nie ma znaczenia, które konkretnie warunki z \mathbb{T}_k zostaną do niego wybrane. Tu wybieraliśmy początkowe warunki z k -tego poziomu, ale wybierając je w inny sposób można uzyskać dodatkowe właściwości związane np. z postaciami interpretacji ρ_A . Jednak zastosowana tu metoda, wydając się najbardziej naturalna, ma tę ciekawą cechę, że interpretacje metazbioru ρ_A odpowiadają α -przekrojom zbioru rozmytego A , czego dowodzi poniższy lemat. Dla uproszczenia notacji przyjmijmy, że dziedziną zbioru rozmytego A jest jakaś skończona liczba porządkowa m , tzn. $X = \{0, \dots, m-1\}$. Dzięki temu odwzorowanie $i_X : \text{dom}(\rho_A) \mapsto X$ będzie po prostu funkcją odwrotną do izomorfizmu kanonicznego: $i_X(\tilde{m}) = m$ (def. 14).

LEMAT 6.1.1. Niech A będzie zbiorem rozmytym w przestrzeni X , która jest pewną skończoną liczbą porządkową $X = m = \{0, \dots, m-1\}$, a funkcja przynależności μ_A przyjmuje wartości wymierne o mianowniku postaci 2^k dla pewnego $k \in \mathbb{N}$. Niech $p_j^k \in \mathbb{T}_k$ dla $0 \leq j < 2^k$ będą kolejnymi elementami k -tego poziomu drzewa \mathbb{T} (indeksowanymi zgodnie z porządkiem poziomu). Niech ρ_A będzie następującym metazbiorem:

$$\rho_A = \{ \langle \tilde{n}, p_j^k \rangle : 0 \leq n < m \wedge 0 \leq j < \mu_A(n) \cdot 2^k \} .$$

¹Porządek węzłów na każdym poziomie jest „od lewej do prawej”, por. rys. 3.1 ze str. 11.

Jeśli C_i jest gałęzią zawierającą jakiś element $p_i^k \in \mathbb{T}_k$ dla pewnego $i < 2^k$, to interpretacja $\text{int}(\rho_A, C_i)$ jest α -przekrojem zbioru rozmytego A , dla $\alpha = (i+1)/2^k$. Ponadto, każdy α -przekrój zbioru rozmytego A jest równy jakiejś interpretacji metazbioru ρ_A .

Dowód. Na początek zauważmy, że różniące się między sobą α -przekroje są wśród tych postaci $S_{i/2^k} = \{x \in X : \mu_A(x) \geq i/2^k \wedge 0 < i \leq 2^k\}$. Każdy inny przekrój jest równy któremuś z nich.

Niech $p_i^k \in \mathbb{T}_k$, a C_i niech będzie dowolną gałęzią zawierającą p_i^k . Każda taka gałąź daje identyczną interpretację metazbioru ρ_A :

$$p_i^k \in C'_i \wedge p_i^k \in C''_i \rightarrow \text{int}(\rho_A, C'_i) = \text{int}(\rho_A, C''_i) .$$

Rzeczywiście, elementy potencjalne ρ_A są metazbiorami kanonicznymi, więc z faktu 3.3.3 wynika, że ich interpretacje są zawsze takie same, niezależnie od gałęzi. Z kolei przeciwdziedzina ρ_A jest całkowicie zawarta w \mathbb{T}_k , zatem nie zawiera ona warunków poniżej p_i^k , które mogą wpłynąć na postać interpretacji.

Z definicji ρ_A wynika, że jego interpretacje są zbiorami liczb porządkowych. Niech więc $n \in \text{int}(\rho_A, C_i)$. Zgodnie z definicją metazbioru ρ_A para $\langle \check{n}, p_i^k \rangle$ wchodzi w jego skład. To oznacza, że musi być $i < l_A^k(i_X(\check{n})) = l_A^k(n)$, czyli $i/2^k < \mu_A(n)$. Ponieważ μ_A przyjmuje wyłącznie wartości wymierne o mianowniku 2^k , to mamy również $(i+1)/2^k \leq \mu_A(n)$. Zatem n należy do α -przekroju zbioru rozmytego A dla $\alpha = (i+1)/2^k$, a w związku z tym $\text{int}(\rho_A, C_i) \subset S_{(i+1)/2^k}$.

Niech teraz $n \in S_{(i+1)/2^k}$ dla $0 \leq i < 2^k$. Oznacza to, że $\mu_A(n) \geq (i+1)/2^k$, czyli równoważnie $\mu_A(n) > i/2^k$. Zatem $i < \mu_A(n) \cdot 2^k$, i zgodnie z jego definicją metazbiór ρ_A zawiera parę $\langle \check{n}, p_i^k \rangle$. Jeśli $C_i \ni p_i^k$ jest gałęzią, to $n \in \text{int}(\rho_A, C_i)$ i ostatecznie $S_{(i+1)/2^k} \subset \text{int}(\rho_A, C_i)$. \square

Dodatkowo odnotujmy, że jeśli C_0 jest jakąś gałęzią przechodzącą przez p_0^k (np. lewą-skrajną), to $\text{int}(\rho_A, C_0) = \text{supp}(A)$ jest nośnikiem zbioru rozmytego A . Jeśli C_{2^k-1} jest gałęzią zawierającą $p_{2^k-1}^k$ (np. prawą-skrajną), to $\text{int}(\rho_A, C_{2^k-1}) = \text{kern}(A)$ jest jądrem zbioru rozmytego A .

Łatwo sprawdzić, że relacje równości i zawierania dla zbiorów rozmytych są zgodne z ich odpowiednikami dla metazbiorów. Również standardowe operacje sumy i części wspólnej pokrywają się, co teraz wykażemy. Dla uproszczenia notacji przyjmijmy, że zbiory rozmyte będące argumentami operacji mają wspólną przestrzeń (dziedzinę), która jest skończoną liczbą porządkową.

LEMAT 6.1.2. Niech A, B będą zbiorami rozmytymi o funkcjach przynależności μ_A i μ_B przyjmujących wartości wymierne o mianowniku postaci 2^k , których dziedziną jest skończona liczba porządkowa m . Dla metazbiorów

$$\begin{aligned} \rho_A &= \{ \langle \check{n}, p_j^k \rangle : n \in m \wedge 0 \leq j < 2^k \cdot \mu_A(n) \} , \\ \rho_B &= \{ \langle \check{n}, p_j^k \rangle : n \in m \wedge 0 \leq j < 2^k \cdot \mu_B(n) \} , \\ \rho_{A \cup B} &= \{ \langle \check{n}, p_j^k \rangle : n \in m \wedge 0 \leq j < 2^k \cdot \max(\mu_A(n), \mu_B(n)) \} , \\ \rho_{A \cap B} &= \{ \langle \check{n}, p_j^k \rangle : n \in m \wedge 0 \leq j < 2^k \cdot \min(\mu_A(n), \mu_B(n)) \} , \end{aligned}$$

zachodzą następujące równości:

$$\rho_{A \cup B} = \rho_A \cup \rho_B , \quad (6.1.7)$$

$$\rho_{A \cap B} = \rho_A \cap \rho_B . \quad (6.1.8)$$

Dowód. Niech $\langle \check{n}, p_j^k \rangle \in \rho_{A \cup B}$. Z definicji $\rho_{A \cup B}$ jest to równoważne alternatywie: $j < 2^k \cdot \mu_A(n)$ lub $j < 2^k \cdot \mu_B(n)$, czyli $\langle \check{n}, p_j^k \rangle \in \rho_A$ lub $\langle \check{n}, p_j^k \rangle \in \rho_B$. To z kolei oznacza, że $\langle \check{n}, p_j^k \rangle \in \rho_A \cup \rho_B = \rho_A \cup \rho_B$ (def. 5.3.1), co dowodzi (6.1.7).

Dla dowodu (6.1.8) zauważmy najpierw, że dla $n < m$ (por. def. 3.2.5):

$$\begin{aligned} \rho_A[\check{n}] &= \{ p_j^k : j < 2^k \cdot \mu_A(n) \} , \\ \rho_B[\check{n}] &= \{ p_j^k : j < 2^k \cdot \mu_B(n) \} , \\ \rho_{A \cap B}[\check{n}] &= \{ p_j^k : j < 2^k \cdot \min(\mu_A(n), \mu_B(n)) \} . \end{aligned}$$

Łatwo widzieć, że (por. def. 5.3.3 i 5.3.4):

$$\begin{aligned} \rho_{A \cap B}[\check{n}] &= \rho_A[\check{n}] \cap \rho_B[\check{n}] , \\ \rho_A[\check{n}] \bowtie \rho_B[\check{n}] &= \rho_A[\check{n}] \cap \rho_B[\check{n}] , \\ (\rho_A \cap \rho_B)[\check{n}] &= \rho_A[\check{n}] \bowtie \rho_B[\check{n}] . \end{aligned}$$

Zatem $\langle \check{n}, p_j^k \rangle \in \rho_{A \cap B}$ wtedy i tylko wtedy, gdy $\langle \check{n}, p_j^k \rangle \in \rho_A \cap \rho_B$. \square

Zwróćmy uwagę na fakt, iż równości (6.1.7), (6.1.8) w tezie lematu są ostre, a nie warunkowe. Zatem metazbiory po obu stronach równości są identyczne jako zbiory, a nie tylko w przybliżeniu równe jako metazbiory.

6.2 Reprezentacje w \mathfrak{M}^1

Opisany sposób reprezentacji zbiorów rozmytych przy pomocy metazbiorów klasy \mathfrak{M}^1 ma przede wszystkim znaczenie dla komputerowych implementacji ze względu na ograniczenia co do wartości funkcji przynależności oraz skończoności dziedziny. Jak wiadomo, w świecie komputerów tego rodzaju ograniczenia nie są istotne, gdyż są powszechnie spełnione. Można jednak tę metodę uogólnić na zbiory rozmyte z dowolną nieskończoną dziedziną i funkcją przynależności przyjmującą dowolne wartości z przedziału jednostkowego.

Można ją również uogólnić na intuicjonistyczne zbiory rozmyte [30], tzn. oprócz stopnia przynależności i nieprzynależności dodatkowo reprezentować tzw. intuicjonistyczny indeks rozmyty (margines wahania). Dla zbiorów rozmytych stopień nieprzynależności jest dopełnieniem stopnia przynależności. Warunek ten nie musi być spełniony w intuicjonistycznych zbiorach rozmytych [3], gdyż oprócz stopni przynależności i nieprzynależności reprezentowanych przez odpowiednie funkcje, uwzględniają one jeszcze margines wahania, będący dopełnieniem do jedności sumy tych stopni.

Podobnie można reprezentować zbiory rozmyte typu 2, gdzie wartościami funkcji przynależności są przedziały zawarte w odcinku jednostkowym.

Reprezentacje intuicjonistycznych zbiorów rozmytych oraz dowolnych zbiorów rozmytych typu 1 i typu 2 opierają się na metazbiorach pierwszego poziomu (klasy \mathfrak{M}^1), które jednak nie muszą być dziedzicznie skończone. Ze względu na charakter niniejszej pracy, która koncentruje się wokół zastosowań informatycznych, pominiemy tu dyskusję pozostałych reprezentacji. Mają one przede wszystkim znaczenie teoretyczne.

Rozdział 7

Implementacja

Pierwsza, eksperymentalna implementacja biblioteki operacji na metazbiorach została wykonana w języku Java. Wybór Javy jako języka pierwszej implementacji podyktowany był kilkoma czynnikami.

Po pierwsze — przenośność umożliwiająca uruchamianie aplikacji korzystających z biblioteki w różnych środowiskach systemowych bez potrzeby utrzymywania osobnych, dedykowanych wersji kodu. Ma to szczególne znaczenie w przypadku aplikacji korzystających z graficznego interfejsu użytkownika, takich jak znajdujące się na załączonej płytce, który zwykle jest oparty na natywnych bibliotekach. Tak więc Java, mimo wad wymienionych poniżej, świetnie nadaje się na „szkolną” wersję implementacji, która ma głównie służyć do wykonywania prezentacji i popularyzacji koncepcji metazbiorów.

Po drugie — relatywnie duża efektywność wygenerowanego kodu w porównaniu z innymi językami wysokiego poziomu o podobnych cechach. Po trzecie — łatwość wykonania implementacji i jej prostota, a co najważniejsze — możliwość uniknięcia subtelnych błędów (np. związanych z zarządzaniem pamięcią), typowych dla takich języków jak C czy C++, co przyspiesza doprowadzenie implementacji do stanu używalności i zwiększa zaufanie do działającego kodu. Proces testowania i usuwania błędów z kodu napisanego w C/C++ jest znacznie bardziej żmudny i długotrwały niż w Javie. Po czwarte wreszcie — możliwość udostępnienia aplikacji opartych na metazbiorach w formie apletów w witrynach internetowych, co przyczyni się do spopularyzowania idei metazbiorów i umożliwi bliższe zapoznanie się z nią szerokiemu gronu potencjalnych odbiorców.

Mimo tych zalet Java nie nadaje się do wykonania w pełni profesjonalnej implementacji, ze względu na niższą efektywność wynikowego kodu niż możliwa do osiągnięcia w C czy C++. W szczególności dyskwalifikującą rolę odgrywa brak możliwości optymalizacji kodu wynikowego oraz brak dostępu do niskopoziomowych struktur maszyny, na których będzie oparta docelowa implementacja wykonana właśnie w C bądź C++.

Zanim to jednak nastąpi, planuje się wykonanie referencyjnej implementacji w języku wysokiego poziomu, umożliwiającym dowiedzenie częściowej poprawności zastosowanych algorytmów i upewnienie się co do niezawodności gotowego kodu. Dobrymi kandydatami są języki z rodziny ML (tj. – *nomen omen* – Meta Language). Ani Java, ani C/C++ nie nadają się do wykonania tego zadania ze względu na specyficzne właściwości tych języków, oraz brak odpowiednich narzędzi do formalnej weryfikacji kodu.

Zatem kolejnym krokiem będzie implementacja odniesienia, nastawiona na niezawodność i poprawność. Dopiero potem zostanie wykonana docelowa implementacja nastawiona na efektywność, gdzie w ramach testów porównawczych zostanie wykorzystana implementacja odniesienia.

Bieżąca, implementacja jest kompromisem pomiędzy efektywnością a prostotą i czytelnością kodu źródłowego. Zastosowane algorytmy są dość szybkie i mają niską złożoność, jednak niektóre ogólne założenia (np. reprezentacja warunków jako napisy, każdy znak na dwóch bajtach) sprawiają, że nie może ona być optymalna. Początkowo nadrzędnym celem była prostota zastosowanych rozwiązań, co miało przede wszystkim ułatwić późniejszą analizę kodu źródłowego, jak również pozytywnie wpłynąć na niezawodność i poprawność biblioteki. W związku z tym – poza przypadkami, kiedy efektywność miała szczególnie duże znaczenie – kod implementowanych operacji jest maksymalnie bliski matematycznym formułom występującym w definicjach i twierdzeniach podanych w rozdziale 5. Ponieważ jest on bezpośrednim przełożeniem na język programowania matematycznych koncepcji, to zwykle nie wymaga też szczegółowych komentarzy ani szerszego opisu.

Z drugiej strony, zastosowano wiele prostych optymalizacji kodu, które nie tylko go nie zaciemniają, jak to ma zwykle miejsce w takich sytuacjach, ale przeciwnie – zwiększają jego czytelność. Przykładem mogą być właściwości metazbiorów takie jak moc dziedziny, moc przeciwdziedziny, poziom metazbioru, czy moc jako liczba par uporządkowanych, które są sprawdzane przy testowaniu równości (w sensie tożsamości) zanim przejdzie się do porównania zawartości metazbiorów. Innym przykładem może być przechowywanie raz utworzonych obiektów reprezentujących warunki w tablicy mieszającej pełniącej rolę pamięci podręcznej, w celu ich ponownego wykorzystania. Dodatkowo, obiekty te leniwie wyliczają i zapamiętują odniesienia do swoich potomków, sąsiadów i rodzica, co przyspiesza późniejsze odwołania do nich.

W kodzie źródłowym, zwłaszcza w bardziej zawiłych jego partiach, zostały zastosowane asercje. Są to tożsamości logiczne i algebraiczne, które muszą być spełnione w trakcie działania programu aby umożliwić jego kontynuację. Niepełniona asercja powoduje przerwanie wykonania programu, o ile testowanie asercji zostało uaktywnione podczas uruchamiania. Asercje są umieszczane zwykle przy wyjściach z metod w celu sprawdzenia poprawności obliczonych wyników. Przykładem może być metoda znajdująca zbiór elementów maksymalnych w zbiorze elementów drzewa binarnego, która przed wyjściem sprawdza między innymi, czy obliczony zbiór jest antylańcuchem (zbiór elementów maksymalnych musi być antylańcuchem). Ponieważ asercje wydłużają czas wykonania i zwiększają zużycie pamięci nie produkując przy tym żadnych wyników, standardowo nie są one wykonywane podczas normalnej pracy maszyny wirtualnej Javy. Aby je aktywować należy użyć opcji `-ea` podczas jej uruchamiania.

Zastosowanie asercji zwiększa zaufanie do kodu, nie stanowi jednak ostatecznej weryfikacji jego poprawności, gdyż formuły w nich użyte nie opisują w pełni (bo nie mogą¹) wszystkich jego istotnych aspektów. Dlatego biblioteka była gruntownie i długotrwanie testowana w różnych środowiskach systemowo-sprzętowych, na losowych danych różnych rozmiarów. Sprawdzane były między innymi tożsamości algebry Boole'a oraz związki pomiędzy podzbiarami drzewa

¹Ze względu na niebezpieczeństwo wzajemnie rekurencyjnych wywołań. Asercje często opisują wyniki produkowane przez metody poprzez wywołania innych metod.

binarnego, takimi jak splot, baza dopełnienia czy zbiór elementów maksymalnych, produkowanymi przez procedury biblioteki. Od czasu usunięcia przypadkowych błędów typograficznych nie zaobserwowano żadnych nieprawidłowości.

7.1 Opis biblioteki

Podstawowymi elementami biblioteki są klasy pakietu `ms`, reprezentujące trzy kategorie metazbiorów. Klasą nadrzędną jest `MetaSet`, której obiekty są metazbiorami (oczywiście, są one dziedzicznie skończone). Z niej wyprowadzono klasę `FirstOrderMetaSet` reprezentującą metazbiory pierwszego poziomu, z której z kolei wyprowadzono klasę `CanonicalMetaSet` reprezentującą metazbiory kanoniczne.

Ważną cechą implementacji jest fakt, iż obiekty powyższych klas są niemodyfikowalne, tzn. nie można dodać ani usunąć elementu z metazbioru reprezentowanego przez obiekty tych klas. Ma to na celu uniknięcie przypadkowych pomyłek programistycznych w aplikacjach korzystających z biblioteki. Nowe obiekty tworzy się podając zawartość metazbiorów w postaci listy par uporządkowanych konstruktorom.

Drzewo binarne jako takie nie jest reprezentowane przez żadną klasę, bo nie ma takiej potrzeby. Jego wierzchołki są obiektami klasy `ms.btree.Condition`, która udostępnia metody służące do porównywania warunków i inne. Obiektów typu `Condition` nie tworzy się konstruktorem, lecz fabrykatorem (statyczną metodą `Condition.fromName(String)` lub `Condition.fromLevel(int, int)`), który zwraca odniesienie do przechowywanej w pamięci podręcznej w jednym egzemplarzu instancji. Dzięki temu unikamy nadmiernego zużycia pamięci na skutek wielokrotnego powoływania nowych obiektów reprezentujących warunki.

Operacje na zbiorach wierzchołków drzewa binarnego, które stanowią podstawę dla określenia relacji mnogościowych i operacji algebraicznych, zostały zaimplementowane w klasie `ConditionSet` pakietu `ms.btree`. Zawiera ona między innymi implementacje algorytmów obliczających splot, bazę dopełnienia, zbiór elementów maksymalnych, testujących pokrywanie, itd. Niektóre operacje są zaimplementowane na kilka sposobów, przy użyciu różnych algorytmów, z których jedne mogą w specyficznych sytuacjach okazać się bardziej efektywne niż pozostałe. W szczególności chodzi o testowanie rozpinania (metody `spreads_*(...)`) i obliczanie bazy dopełnienia (metody `maxInc_*`). Domyślne wersje tych algorytmów są wywoływane przez metody `maxInc()` oraz `spreads(Condition)`.

Podstawowa dla zastosowań biblioteki klasa `FirstOrderMetaSet` udostępnia metody testujące relacje mnogościowe i realizujące operacje algebraiczne na metazbiorach pierwszego poziomu w sposób przedstawiony w rozdziale 5. W szczególności

- metoda `add(FirstOrderMetaSet)` zwraca sumę,
- metoda `intersect(FirstOrderMetaSet)` zwraca przecięcie,
- metoda `subtract(FirstOrderMetaSet)` zwraca różnicę,

obiekty, na rzecz którego metoda została wywołana i argumentu. Natomiast

- metoda `condSubset(FirstOrderMetaSet, Condition)` testuje zawieranie warunkowe,

- metoda `condEquals(FirstOrderMetaSet, Condition)` testuje równość warunkową

adresata komunikatu i argumentu. Należenie dla metazbiorów pierwszego poziomu jest zaimplementowane w klasie `CanonicalMetaSet` w postaci metody `belongsTo(FirstOrderMetaSet, Condition)`, która testuje należenie warunkowe adresata komunikatu do argumentu. Dodatkowo, klasa `MetaSet` udostępnia metodę `interpret(Branch)`, która zwraca interpretację tego metazbioru zadaną przez gałąź (będącą obiektem klasy `ms.btree.Branch`) przekazaną jako argument. Wynik jest obiektem klasy `ms.ZFSet` reprezentującej klasyczne zbiory (Zermelo-Fraenkla).

Wewnętrznie, metazbiór jako obiekt typu `MetaSet` zawiera jako atrybut tablicę mieszającą typu `HashMap<MetaSet, Image>`, która kojarzy elementy potencjalne tego metazbioru z ich obrazami. Obrazy są obiektami wewnętrznej klasy `Image` reprezentującej niemodyfikowalne zbiory warunków. Konstruktory klasy `MetaSet` – i siłą rzeczy również jej podklas – inicjują atrybuty reprezentujące dziedzinę i przeciwdziedzinę tego metazbioru, będące obiektami wewnętrznych klas `Domain` i `Range`. Klasy te są niemodyfikowalnymi zbiorami obiektów typu `MetaSet` i `Condition`, odpowiednio.

7.2 Dokumentacja

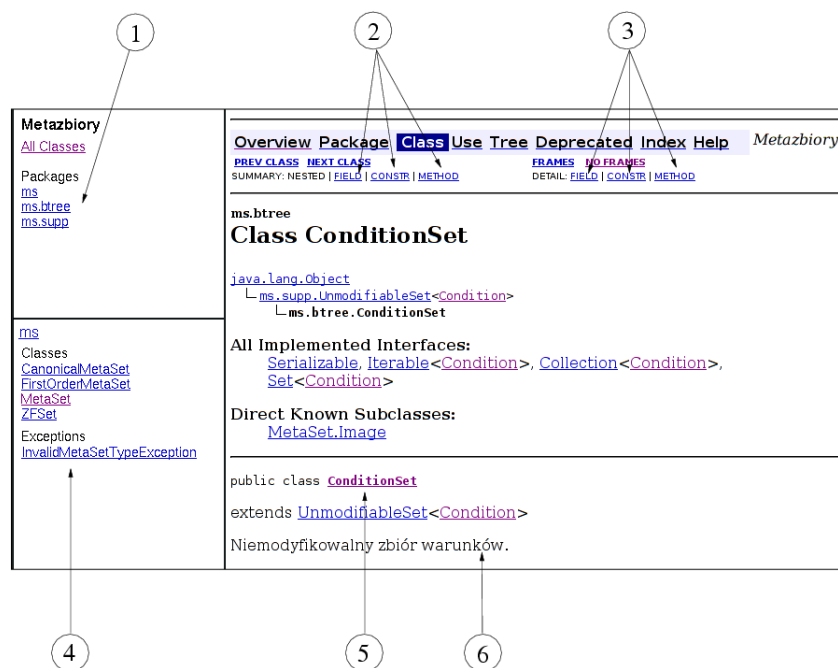
Zestawienie pakietów i klas biblioteki wraz z opisem zawiera dokumentacja *java-doc* znajdująca się na załączonej płytce. Kod źródłowy zawiera również komentarze wyjaśniające szczegóły niektórych rozwiązań. Dokumentację otwiera się w przeglądarce WWW wybierając plik `index.html` z katalogu ją zawierającego. Prezentuje ona opisy poszczególnych pakietów, klas i ich składowych (konstruktorów, atrybutów, metod). Umożliwia również przeglądanie kodu źródłowego bezpośrednio z poziomu przeglądarki. Sposób posługiwania się dokumentacją wyjaśnia rys. 7.1.

Okno przeglądarki podzielone jest na trzy części. W lewym-górnym rogu (1)² widzimy listę pakietów, w które pogrupowane są klasy. Po kliknięciu nazwy pakietu w lewym-dolnym rogu (4) zobaczymy zawartość tego pakietu. Klikając nazwę klasy w prawej części okna (6) zobaczymy jej opis (na rys. 7.1 widoczny tylko częściowo). Łącze z nazwą klasy (5) w jej głównym opisie prowadzi do strony z kodem źródłowym klasy (komentarze, w szczególności polskie znaki są tu kodowane w systemie iso8859-2). Dostęp do skróconego opisu pól, konstruktorów i metod uzyskamy klikając jedno z łączy (2). Do pełnego opisu prowadzą łączy (3). Dokumentacja off-line biblioteki jest zintegrowana z dokumentacją on-line standardowej biblioteki Javy, tzn. łączy do opisów standardowych klas (jak np. `String`) prowadzą do witryny `java.sun.com`.

7.3 Dodatkowe aplikacje

Biblioteka klas reprezentujących metazbiory posłużyła do stworzenia dwóch aplikacji. Pierwsza z nich ma charakter przede wszystkim edukacyjny i służy do wizualizacji metazbiorów. Druga jest przykładem zastosowania teorii do rozwiązania praktycznego problemu.

²Cyfry bądź literki w nawiasach odnoszą się do oznaczeń na rysunkach.



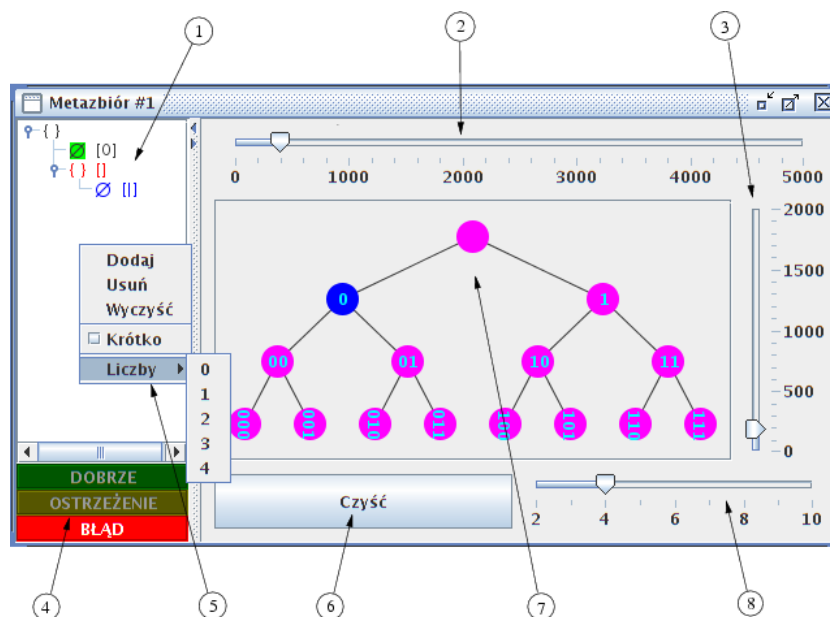
Rysunek 7.1: Dokumentacja kodu źródłowego biblioteki

7.3.1 Pulpit metazbiorów

W celu lepszego zrozumienia koncepcji i budowy metazbiorów przygotowano aplikację prezentującą graficznie ich strukturę. Umożliwia ona wizualne definiowanie metazbiorów, wykonywanie operacji algebraicznych oraz testowanie relacji mnogościowych. Stworzone w tym programie metazbiory można zapisywać do plików w standardowym, uniwersalnym formacie, który pozwala na ich odczytanie w innych aplikacjach (są to serializowane obiekty Javy; klasy `MetaSet` i `Condition` implementują interfejs `Serializable`).

Interfejs graficzny programu przedstawia rys. 7.4. Składa się on z trzech elementów: pulpitu zawierający okna wewnętrzne reprezentujące metazbiory (po prawej), pasek narzędzi umożliwiający wykonywanie operacji i określanie ich argumentów (u góry), oraz panel biblioteczny przechowujący zapisane wcześniej w plikach metazbiory (po lewej). Na początku tworzone jest jedno okno wewnętrzne reprezentujące metazbiór pusty, który następnie rozbudowuje się dodając elementy potencjalne oraz określając ich obrazy będące zbiorami warunków. W trakcie działania programu definiujemy nowe metazbiory i wykonujemy operacje na nich.

Rysunek 7.2 przedstawia pojedyncze okno wewnętrzne służące do definiowania metazbioru. Składa się ono z dwóch głównych obszarów. Lewy ukazuje strukturę hierarchii należenia w formie drzewa (1). Prawy panel zawiera drzewo binarne (7) z zaznaczonymi na niebiesko węzłami należącymi do obrazu bieżącego, podświetlonego zielonym kwadratem, elementu potencjalnego z lewego panelu. Warunki dodaje się do obrazu zaznaczając je lewym klawiszem myszki, a usuwa prawym klawiszem. Przycisk (6) służy do wyzerowania zawartości obrazu (usuwa z niego wszystkie warunki). Widok drzewa binarnego można dosto-



Rysunek 7.2: Pojedyncze okno wewnętrzne reprezentujące metazbiór

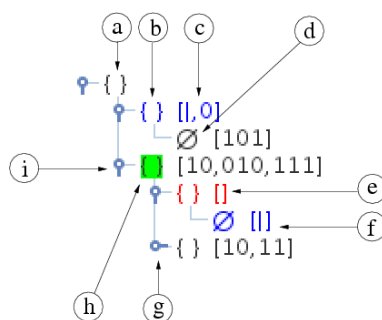
sowywać suwakami. Górny (2) i prawy (3) służą do zmiany rozmiaru wirtualnej płaszczyzny, na której wykreślane jest drzewo. Dzięki temu można powiększyć lub przybliżyć jakiś jego fragment. Dolny suwak (8) reguluje widoczną liczbę poziomów drzewa. Maksymalną liczbą poziomów jest 10, gdyż więcej praktycznie nie da się zobaczyć na monitorze. Poza tym, zużycie zasobów maszyny rośnie wykładniczo wraz ze wzrostem liczby poziomów, negatywnie wpływając na interakcję.

Początkowo, kiedy nowe okno wewnętrzne reprezentuje zbiór pusty, edycja warunków w prawym panelu jest niemożliwa, gdyż aktywny metazbiór z lewego panelu nie jest elementem potencjalnym. Warunki można zaznaczać dla metazbiorów, które są elementami potencjalnymi innych metazbiorów. Zmianę aktywnego, edytowanego metazbioru w panelu hierarchii należenia wykonuje się kliknięciem myszką w element, który ma być edytowany. Uzyskuje on wtedy zielone tło.

Najpierw trzeba jednak takie elementy potencjalne utworzyć i umiejscowić w hierarchii należenia. Służy do tego menu rozwijalne (5), wyzwalane prawym klawiszem myszy w obszarze lewego panelu. Menu otwieramy ustawiając kursor nad metazbiorem, do którego będziemy dodawać element potencjalny. Następnie wybieramy pierwszą pozycję *Dodaj*, która dodaje metazbiór pusty jako element potencjalny. Można też wybrać jedną z pozycji podmenu *Liczby*, które zawiera gotowe kanoniczne metazbiory będące odpowiednikami liczb porządkowych, aby dodać taki metazbiór do dziedziny zaznaczonego metazbioru. Pozycja *Usuń* tego menu usuwa element pod kursorem, a pozycja *Wyczyść* usuwa wszystkie metazbiory z lewego panelu. Pozycja *Krótko* przełącza pomiędzy skróconym i pełnym opisem widocznych metazbiorów.

Dodając lub usuwając warunki do obrazu aktywnego metazbioru w panelu hierarchii należenia, można spowodować, że utworzony zbiór nie będzie meta-

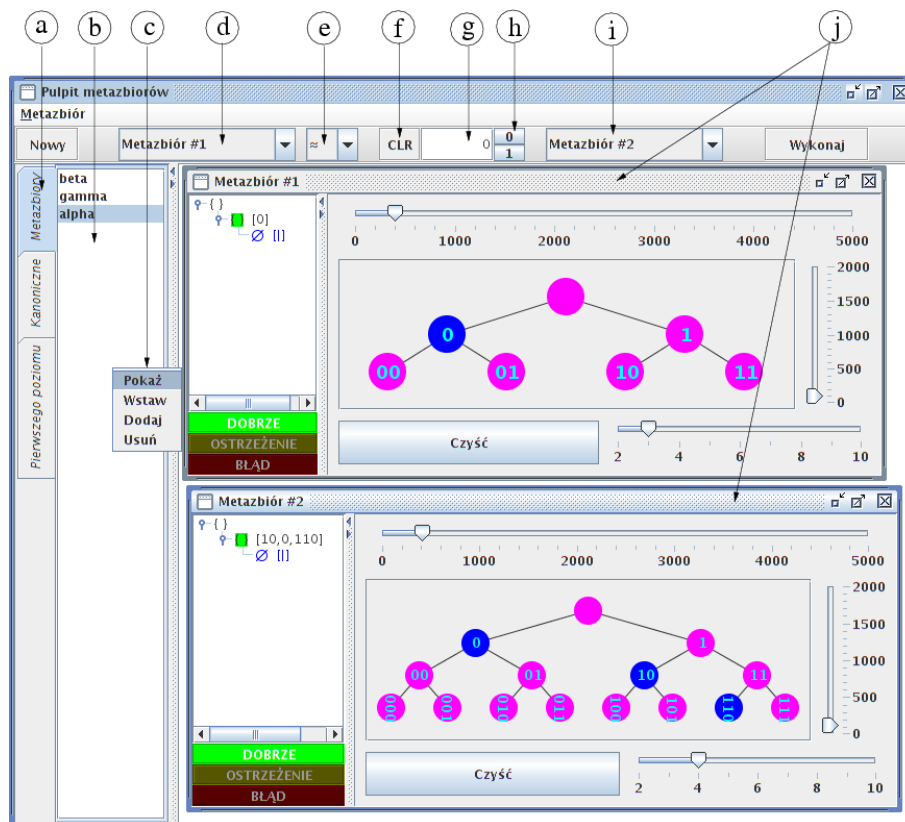
zbiorem z dwóch powodów. Po pierwsze jakiś element potencjalny w hierarchii należenia może mieć pusty obraz w metazbiorze stojącym bezpośrednio wyżej w hierarchii na skutek usunięcia wszystkich elementów obrazu. Po drugie, może się okazać, że gdzieś w hierarchii należenia pojawiły się jako elementy dziedziny jednego metazbioru dwa identyczne metazbiory posiadające do tego identyczne obrazy. Pierwsza sytuacja jest ewidentnym błędem sprzecznym z definicją i nie wolno do niej dopuszczać. Druga w zasadzie nie jest sprzeczna z definicją, bo z punktu widzenia teorii mnogości dwa identyczne zbiory są traktowane jako jeden, ale takie sytuacje powodują zamieszanie, i z tego powodu należy ich unikać. Aby pomóc ustrzec się przed tego rodzaju błędami, w lewym-dolnym rogu znajdują się kolorowe etykiety (4) informujące o stanie struktury definiowanego metazbioru przy pomocy świetlnych komunikatów. Światło czerwone oznacza, że jakiś element potencjalny ma pusty obraz i należy go uzupełnić. Światło żółte oznacza, że gdzieś w hierarchii należenia pojawił się metazbiór zawierający dwa identyczne elementy potencjalne, co jest niewłaściwe i wymaga modyfikacji jednego z nich. Światło zielone oznacza, że metazbiór jest poprawnie zbudowany. Struktura metazbioru jest kontrolowana po każdorazowym dodaniu elementu potencjalnego albo modyfikacji obrazu.



Rysunek 7.3: Drzewo hierarchii należenia

Przyjrzyjmy się teraz dokładniej informacji prezentowanej w panelu hierarchii należenia na przykładzie rys. 7.3. Korzeniem drzewa należenia (a) jest metazbiór reprezentowany przez całe okno wewnętrzne. Po prawej stronie, w każdym wierszu z wyjątkiem pierwszego, w nawiasach kwadratowych (c) prezentowane są zbiory warunków. Są to obrazy metazbiorów zaznaczonych symbolicznie po lewej stronie przed nawiasem otwierającym, albo jako zbiór pusty \emptyset (d), który jest liściem w drzewie należenia, albo jako para nawiasów klamrowych $\{ \}$ symbolizująca węzeł – zbiór z niepustą zawartością (b). Ta zawartość jest reprezentowana jako poddrzewo hierarchii należenia, analogicznie do całego drzewa. Wiersze z pustymi obrazami są zaznaczane kolorem czerwonym (e), natomiast wiersze zawierające korzeń drzewa binarnego w obrazie są zaznaczane na niebiesko (f). Takie elementy potencjalne należą w sposób „pełny” (pod warunkiem $\mathbb{1}$), dlatego są wyróżnione. Sam korzeń jest tu zaznaczany symbolem „|”. Aktywny metazbiór, którego obraz jest widoczny w panelu widoku drzewa binarnego, jest podświetlony na zielono (h). Zmianę aktywnego metazbioru wykonuje się kliknięciem w symbol odpowiedniego zbioru. Drzewo hierarchii należenia można spłaszczyć w dowolnym miejscu klikając element (i), co da efekt taki jak w węźle (g), który wydaje się być liściem, chociaż nim nie jest, bo nie jest zbiorem pustym

(jest to dodatkowy efekt dostarczony przez użyty tu komponent `JTree` biblioteki `Swing`). Wspomniana wcześniej pozycja *Krótko* menu rozwijalnego ukrywa bądź uwidocznia obrazy poszczególnych elementów w hierarchii przedstawione w nawiasach kwadratowych.



Rysunek 7.4: Pulpit metazbiorów

Przejdźmy teraz do omówienia pozostałych elementów interfejsu graficznego, przedstawionych na rys. 7.4. Okna do edycji metazbiorów tworzy się przyciskiem *Nowy* lub powstają jako wynik operacji algebraicznej wyzwalanej przyciskiem *Wykonaj*. W celu wykonania operacji algebraicznej lub sprawdzenia relacji należy wybrać z list rozwijalnych (d) oraz (i) znajdujących się w pasku narzędziowym okna zawierające metazbiory będące lewym i prawym argumentem. Każde okno ma swoją jednoznacznie identyfikującą nazwę (kolejny numer) wyświetloną na belce (j). Rodzaj operacji wybiera się z listy rozwijalnej (e). Warunki dla relacji warunkowych definiuje się w polu tekstowym (g) jako ciągi zer i jedynek poprzez wpisywanie cyfr z klawiatury bądź klikanie przycisków (h). Do czyszczenia pola tekstowego myszką służy przycisk (f). Wynik operacji algebraicznej jest metazbiorem wyświetlonym w nowym oknie wewnętrznym, natomiast spełnienie relacji jest komunikowane przy pomocy okien dialogowych.

Aplikacja umożliwia zapis do plików oraz odczyt poszczególnych metazbiorów jak również całych sesji zawierających bieżący stan wszystkich okien wewnętrznych. Służą do tego elementy menu *Metazbiór*. W dialogu zapisu pliku

należy wybrać docelowy katalog oraz filtr typu plików z listy rozwijalnej u dołu. Filtr ogranicza widoczne w dialogu typy plików i powoduje automatyczne dodanie właściwego rozszerzenia nazwy pliku. Są następujące filtry, w nawiasach odpowiadające im rozszerzenia: *Metazbiory* (.ms), *Metazbiory pierwszego poziomu* (.fms) oraz *Metazbiory kanoniczne* (.cms). Pliki sesji mają rozszerzenie .mss. Jeśli w katalogu roboczym programu znajdował się podkatalog *libs*, to znajdujące się w nim pliki zapisanych metazbiorów zostaną automatycznie wczytane i udostępnione w panelu bibliotecznym (b). Wczytane metazbiory są pogrupowane ze względu na typ; dostęp do poszczególnych typów umożliwiają zakładki (a). Jeśli katalogu *libs* nie było, to nastąpi próba jego utworzenia. W trakcie działania programu będzie można w nim zapisywać metazbiory przy pomocy menu rozwijalnego (c) wyzwalanego prawym klawiszem myszy w obszarze panelu bibliotecznego. Polecenie *Dodaj* tego menu dodaje do biblioteki metazbiór z aktywnego okna wewnętrznego, natomiast *Usuń* usuwa zaznaczony element biblioteki. Polecenie *Pokaż* wyświetla zaznaczony metazbiór z biblioteki w nowym oknie wewnętrznym, natomiast *Wstaw* wstawia wybrany element biblioteki jako element potencjalny w bieżącym węźle drzewa należenia metazbioru reprezentowanego przez aktywne okno.

7.3.2 Rozpoznawanie znaków

Jako poletko doświadczalne dla zastosowań teorii metazbiorów wybrano problem rozpoznawania pisma, a właściwie znaków. Podstawy teoretyczne dla mechanizmu umożliwiającego reprezentowanie wzorców znaków przy pomocy metazbiorów, oraz relacji ich podobieństwa przy pomocy warunkowego należenia i warunkowej równości, zostały wyłożone w pracy [28] (jednak ze względu na formalne ograniczenia tylko dla pewnego, szczególnego przypadku). Aplikacja implementująca ten mechanizm umożliwia definiowanie złożonych wzorców znaków oraz testowanie dostarczonych próbek pod kątem ich podobieństwa do zdefiniowanego wzorca.

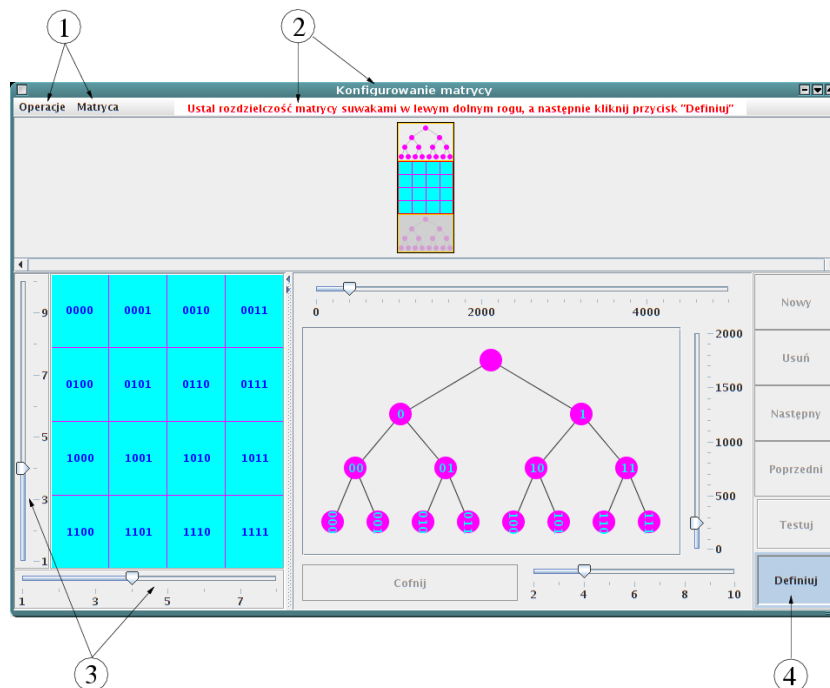
Idea pomysłu jest następująca. Złożony wzorec znaku, składający się z kilku próbek wraz z ocenami ich jakości, jest implementowany jako metazbiór. Testowa próbka znaku jest również reprezentowana jako metazbiór. Można zatem policzyć stopień warunkowego należenia testowej próbki do złożonego wzorca, oraz stopnie warunkowej równości próbki testowej i każdego elementu potencjalnego złożonego wzorca, które reprezentują znaki wchodzące w skład wzorca. Stopnie te interpretuje się jako stopnie podobieństwa próbki testowej do wzorca i jego elementów.

Znaki są reprezentowane na matrycy prostokątnej, w której każdej komórce przyporządkowano warunek z drzewa \mathbb{T} . Dziedzina metazbioru reprezentującego znak składa się z jednego elementu, którym jest \emptyset . Jego obraz w tym metazbiorze składa się z warunków odpowiadających komórkom matrycy zawierającym piksele reprezentowanego znaku. Złożony wzorec znaku składa się z kilku próbek znaków (czyli metazbiorów) z przyporządkowanymi ocenami jakości. Oceny są formułowane jako podzbiory drzewa binarnego będące obrazami elementów potencjalnych odpowiadających próbkom znaków, albo inaczej mówiąc jako stopnie należenia próbek do złożonego wzorca.

Aplikacja umożliwia wizualne definiowanie znaków na matrycy wraz z ich ocenami w drzewie binarnym, oraz dostarczanie próbek testowych. Jako wynik działania dostajemy stopień należenia warunkowego próbki testowej do złożo-

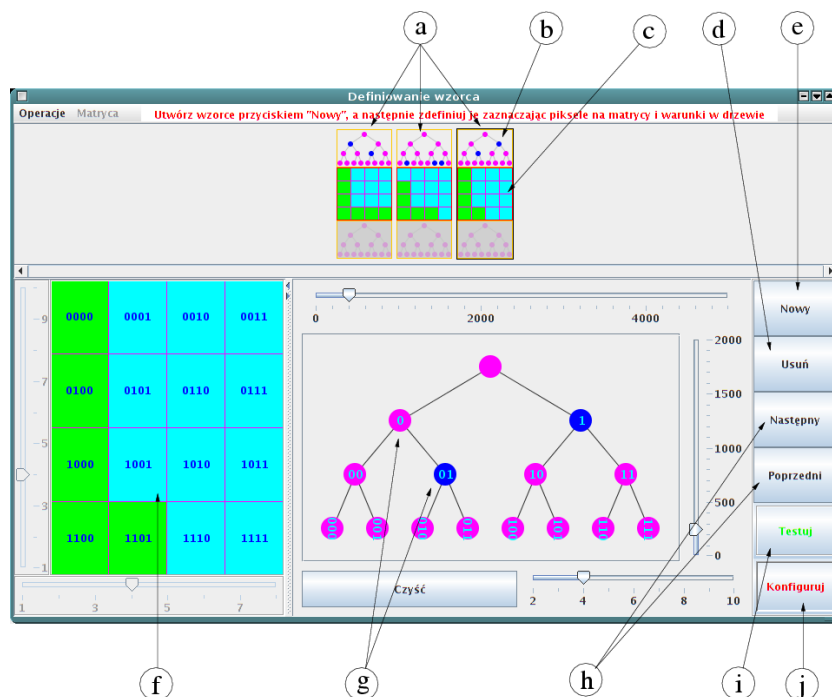
nego wzorca, oraz ciąg stopni równości warunkowych próbki testowej i elementów wzorca.

Praca z programem składa się z trzech kroków. Najpierw należy zdefiniować rozdzielczość matrycy i ewentualnie przyporządkować pikselom warunki. Następnie należy zdefiniować złożony wzorec wraz z ocenami jakości. Ostatnim etapem jest testowanie próbek dające w wyniku stopnie podobieństwa. Poszczególne etapy przedstawiają kolejne ilustracje.



Rysunek 7.5: Konfigurowanie rozdzielczości matrycy

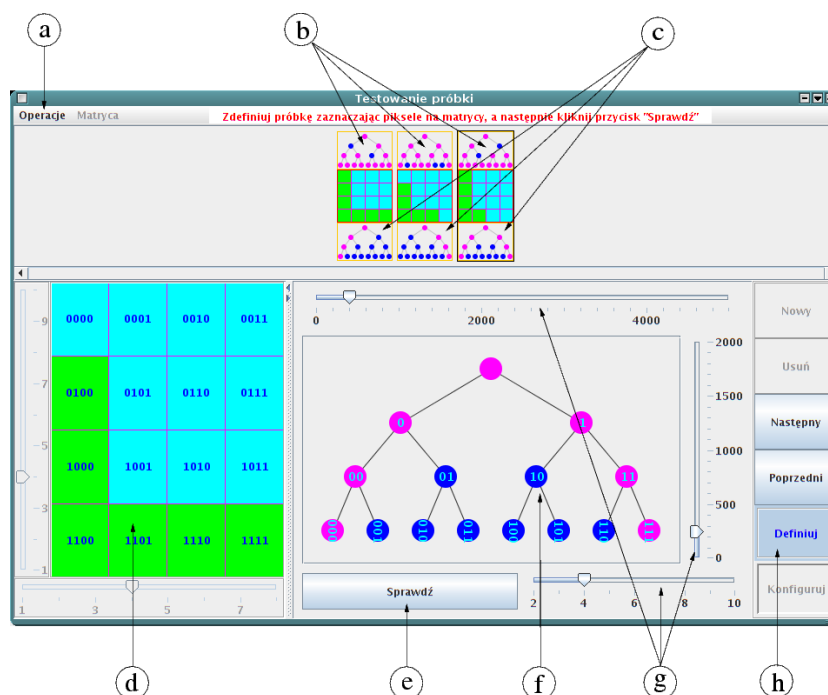
Rysunek 7.5 przedstawia stan interfejsu graficznego w trakcie definiowania matrycy. Na belce i pasku menu (2) wyświetlane są informacje o aktualnym etapie i wskazówki co do kroków, które należy wykonać. Suwakami (3) regulujemy rozdzielczość poziomą i pionową. W polach matrycy wyświetlane są warunki domyślnie im przyporządkowane. Przyporządkowanie to można zmienić przeciągając i upuszczając wierzchołki drzewa do komórek matrycy myszką. Po wstawieniu nowego warunku do matrycy w miejsce starego albo do pustej komórki, wszystkie warunki z nim porównywalne są usuwane z matrycy pozostawiając puste pola, które trzeba wypełnić. Dzieje się tak dlatego, że zbiór warunków zmapowanych na komórki musi być antylańcuchem. Najlepiej jest, jeśli jest on do tego maksymalny. Skonfigurowaną matrycę (tj. ustawioną rozdzielczość wraz z przyporządkowaniem warunków) można zapisać do pliku w celu jej późniejszego odtworzenia wybierając menu *Matryca->Zapisz* (1). Menu *Operacje* zawiera opcję *Nowy*, które w dowolnym miejscu programu przywraca stan początkowy sprzed konfiguracji, powodując równocześnie utratę wszystkich zdefiniowanych danych. Przygotowanie matrycy kończymy przyciskiem *Definiuj* (4) przechodząc w ten sposób do etapu definiowania próbek i złożonego wzorca.



Rysunek 7.6: Definiowanie złożonego wzorca

GUI aplikacji w drugim etapie przedstawia rys. 7.6. Zielone pola na matrycy (f) oznaczają piksele definiowanej nowej próbki znaku. Komórki zaznacza się lewym klawiszem myszy, a odznacza prawym. Niebieskie węzły drzewa binarnego (g) są warunkami z obrazu tej próbki w złożonym wzorcu, czyli tworzą jej ocenę jakościową. Nowe próbki tworzy się przyciskiem *Nowy* (e), a istniejące można usunąć przyciskiem *Usuń* (d). Przyciski *Następny* i *Poprzedni* (h) służą do wyboru bieżącej próbki w celu jej edycji. W górnej części okna aplikacji widzimy miniaturki tworzonych próbek (a). Każda z nich składa się z trzech części, z których górna (b) jest oceną próbki reprezentowanej na matrycy (c). Dolna część będzie prezentować stopień równości na etapie testowania. Kiedy zestaw próbek jest gotowy przechodzimy do ostatniego etapu przyciskiem *Testuj* (i). W każdej chwili można powrócić do etapu konfiguracji przyciskiem *Konfiguruj* (j), tracąc tym samym wszystkie dane.

Ostatnią fazę przedstawia rys. 7.7. Na tym etapie tworzymy próbki testowe zaznaczając ich piksele na matrycy (d) lewym klawiszem myszy i odznaczając prawym, a następnie przyciskiem *Sprawdź* (e) obliczamy stopnie równości próbki testowej i elementów złożonego wzorca, oraz stopień jej należenia do tego wzorca. Stopień należenia jest wyświetlany na panelu drzewa (f), natomiast stopnie równości znajdują się w dolnej części każdej z miniatur (c). Ich górne części (b) pokazują obrazy reprezentowanych przez nie próbek traktowanych jako elementy potencjalne w złożonym wzorcu, czyli stopnie przynależności elementów złożonego wzorca do niego samego. Miniaturki pokazują tylko 4 poziomy drzewa. Jeśli warunki stopni równości znajdują się na niższych poziomach, możemy je zobaczyć zaznaczając myszką odpowiednią miniaturkę. Stopień równości zostanie



Rysunek 7.7: Testowanie próbki

wtedy wyświetlony w głównym panelu drzewa (f), gdzie suwakami (g) możemy regulować widzialny zakres. Przyciskiem *Definiuj* (h) możemy wrócić do definiowania złożonego wzorca kontynuując od momentu przejścia do testowania. Zakończenie działania programu bądź rozpoczęcie nowej sesji następuje opcjami menu *Operacje* (a).

Wyniki działania programu należy interpretować w ten sposób, że większy stopień należenia próbki testowej do wzorca oznacza jej większe podobieństwo. Podobnie, ciąg stopni równości warunkowych pozwala wskazać, do których elementów wzorca testowa próbka jest bardziej podobna, a do których mniej. Jeśli w którymś wynikowym podzbiore drzewa pojawi się korzeń będzie to oznaczało, że dana próbka jest doskonale podobna do elementu wzorca lub całego złożonego wzorca.

W zaprezentowanym mechanizmie jeden element jest na obecną chwilę nie do końca rozpoznany. Chodzi o sposób definiowania ocen próbek przy pomocy zbiorów warunków. Generalnie, warunek z wyższego poziomu oznacza większą wartość, lepszą ocenę próbki. Z dwóch porównywalnych warunków istotną informację wnosi słabszy. Są to konsekwencje lematów 4.1.1 i 4.1.2. Jednak w obrębie jednego poziomu możemy do pewnego stopnia manipulować warunkami będącymi elementami oceny, bez zmiany oceny ilościowej, a modyfikując w ten sposób ocenę jakościową. Konsekwencje takich manipulacji i ich wpływ na końcowy wynik wymykają się na razie próbom uogólnienia. Również związek pomiędzy mapowaniem warunków na komórki matrycy (określonym w pierwszym etapie) a strategią wyboru warunków z danego poziomu jako elementów oceny próbki jest przedmiotem badań.

Należy podkreślić, że zarówno sama koncepcja opisanego mechanizmu, jak i realizująca go aplikacja, ma charakter eksperymentalno-naukowy i nie powinna być traktowana jako komercyjne rozwiązanie problemu rozpoznawania pisma. Głównym celem było sprawdzenie jak teoria metazbiorów sprawuje się w praktycznych zastosowaniach. Eksperyment wypadł pomyślnie. Aplikacja prawidłowo rozpoznaje nie tylko znaki podobne do wchodzących w skład wzorca, ale również wykazuje tendencje do uogólniania wiedzy reprezentowanej przez oceny próbek zdefiniowanego wzorca.

Rozdział 8

Wnioski i zadania na przyszłość

W niniejszej pracy zostały zaprezentowane jedynie podstawy teorii metazbiorów w zakresie niezbędnym do udowodnienia twierdzenia 5.5.9 (algebra Boole'a), które ma fundamentalne znaczenie dla komputerowych reprezentacji metazbiorów. Stopień rozwoju teorii jest jednak daleko bardziej zaawansowany. W szczególności, oprócz wielu istotnych faktów i twierdzeń tu pominiętych, a opublikowanych między innymi w [30], zdefiniowano pojęcie mocy i równoliczności (publikacja w przygotowaniu, [26]) wykazując, że równoliczne metazbiory mają tę samą moc. Prace nad pojęciem metaliczby, która jest mocą metazbioru, wraz z definicjami operacji arytmetycznych są na ukończeniu. Generalnie, rozwój teorii metazbiorów będzie w tym obszarze podążał ścieżką wydeptaną przez teorię zbiorów rozmytych, aby móc z nią w przyszłości skutecznie konkurować lub też ją uzupełniać (por. [33], [16], [15]).

Za główny wynik teoretyczny przedstawiony w rozprawie uważam twierdzenia 5.5.8 i 5.5.9 mówiące, że rodzina metapodzbiorów pierwszego poziomu ustalonego $\delta \in \mathfrak{M}\mathfrak{F}^1$ jest algebrą Boole'a. Nie trzeba chyba tłumaczyć, dlaczego te właściwości są istotne dla pojęcia (meta)zbioru. Bardzo ważną cechą zdefiniowanych operacji jest fakt, że wielokrotne ich wykonywanie nie zwiększa poziomu rozmycia, co ma miejsce w przypadku zbiorów rozmytych.

W rozprawie pominięto bardzo ważne twierdzenia opisujące właściwości metazbiorów dziedzicznie skończonych, które zostaną opublikowane w przygotowywanej pracy [29]. Wśród nich jest zasada trychotomii: jeśli Φ jest formułą (w rozumieniu definicji wymuszania 4.1.1), to albo $\models \Phi$, albo $\models \neg\Phi$, albo $\exists p, q : p \models \Phi \wedge q \models \neg\Phi$. Jest również twierdzenie mówiące, że jeśli jakieś zdanie jest prawdziwe w jakiejś interpretacji, to jest ono wymuszane przez jakiś warunek. W ogólnym przypadku, jeśli metazbiory, o których mówią zdania nie są dziedzicznie skończone, te twierdzenia są nieprawdziwe.

Jak wspomniano w rozdziale 5 (por. przypis na str. 29), założenie dziedzicznej skończoności można pominąć w zaprezentowanych tam tezach. Przede wszystkim chodzi o twierdzenia 5.2.2, 5.2.5 i 5.2.8 mówiące, że nowe sformułowania relacji mnogościowych są poprawne, tzn. zgodne ze zdefiniowanymi w rozdziale 4. Nie ma to oczywiście żadnego znaczenia dla implementacji, gdyż w świecie komputerów mamy do czynienia wyłącznie z bytami skończonymi. Jednak

może mieć znaczenie w rozważaniach teoretycznych, np. przy uogólnianiu zbiorów rozmytych.

Związki ze zbiorami rozmytymi szczególnej postaci, przedstawione w rozdziale 6, można rozszerzyć na dowolne zbiory rozmyte typu 1 oraz i typu 2, oraz na intuicjonistyczne zbiory rozmyte. Na razie nie wiadomo jak daleko idące może być takie uogólnienie, w szczególności jakie operacje i w jakich przypadkach się pokrywają.

Warstwa zastosowań metazbiorów została zaprezentowana bardzo pobieżnie, gdyż jest to obecnie obszar intensywnych badań i w związku z tym nie przyszedł jeszcze czas na formułowanie ogólnych mechanizmów.

Trwają też prace nad zdefiniowaniem wielowartościowej logiki dla języka metazbiorów, opartej na relacji wymuszania. Warunek, który wymusza zdanie, może być traktowany jako jego stopień prawdziwości. Na razie nie wiadomo jednak jak określić reguły wnioskowania.

W najbliższym czasie prace nad metazbiorami będą koncentrować się wokół implementacji i zastosowań teorii. Jak wspomniano w rozdziale 7, planowane jest wykonanie referencyjnej implementacji (prawdopodobnie w języku ML) oraz jej formalna weryfikacja. Następnie zostanie wykonana docelowa, nastawiona na efektywność, implementacja w języku C++. Dopiero wtedy będzie można przystąpić do poważnego stosowania teorii metazbiorów.

Zawartość płyty

Na płycie znajduje się kod źródłowy biblioteki klas Javy reprezentujących metazbiory wraz z dokumentacją, oraz dwie aplikacje bazujące na bibliotece opisane w rozdz. 7. Do ich uruchomienia niezbędna jest poprawnie zainstalowana maszyna wirtualna Javy co najmniej w wersji 6 (JRE 6 lub JDK 6).¹ Do przeglądania dokumentacji potrzebna jest dowolna przeglądarka WWW.

W katalogu głównym na płycie znajdują się trzy podkatalogi: **Biblioteka**, **GUI** i **OCR**, oraz plik `index.html` zawierający łącza do dokumentacji i niezbędne informacje.

Katalog Biblioteka

Zawiera implementację biblioteki klas Javy reprezentujących metazbiory. Podkatalog `src` zawiera kod źródłowy. W podkatalogu `doc` znajduje się dokumentacja, którą można przeglądać po otwarciu pliku `index.html` (z katalogu `doc`) w przeglądarce WWW. Sposób korzystania z dokumentacji opisano w rozdz. 7.2. Archiwum `mslib.jar` zawiera bibliotekę. Skrypty `booltest.sh` (Linux/Unix) i `booltest.bat` (Windows) uruchamiają test tożsamości algebraicznych zaimplementowany w klasie `ms.supp.BoolTest`.

Katalog GUI

Zawiera aplikację służącą do wizualnego definiowania metazbiorów, przedstawioną w podrozdz. 7.3.1. W podkatalogu `src` znajduje się kod źródłowy aplikacji, podkatalog `libs` zawiera bibliotekę metazbiorów wykorzystywaną przez aplikację, a w podkatalogu `doc` znajduje się dokumentacja (plik `doc/index.html`). Archiwum `mslib.jar` zawiera bibliotekę, a skrypty `msgui.sh` i `msgui.bat` służą do uruchamiania aplikacji.

Uwaga: *aplikacja tworzy katalog `libs` w swoim katalogu uruchomionym do przechowywania plików bibliotecznych. Jeśli jest to niemożliwe – tak jak w przypadku uruchamiania z płyty, której system plików jest tylko do odczytu – to próbuje jedynie odczytać pliki metazbiorów przechowywane w tym katalogu (o ile on istnieje; na płycie jest). Zapisywanie nowych metazbiorów do biblioteki będzie w takim przypadku niemożliwe, jednak w dalszym ciągu można zapisywać sesje i metazbiory do plików w innych lokalizacjach za pośrednictwem menu.*

¹Można ją pobrać z witryny <http://java.sun.com/javase/downloads/index.jsp>, aczkolwiek nie powinno to być konieczne, gdyż Java jest obecna we wszystkich nowoczesnych, poprawnie skonfigurowanych systemach.

Katalog OCR

Zawiera aplikację rozpoznającą znaki opisaną w podrozdz. 7.3.2. Uruchamiają ją skrypty `msocr.sh` lub `msocr.bat`, a sama aplikacja znajduje się w archiwum `msocr.jar`.

Indeks symboli

$Q \parallel R$, 32
 $Q \parallel_p R$, 33
 $R \mid p$, 31
 $W \upharpoonright_p$, 33
 $[\dots]$, 10
 $\mathbb{1}$, 9
 \mathfrak{M} , 12
 $\mathfrak{M}\mathfrak{F}$, 15
 $\mathfrak{M}\mathfrak{F}^c$, 30
 $\mathfrak{M}\mathfrak{F}^\perp$, 30
 \mathfrak{M}^c , 14
 \mathfrak{M}^\perp , 30
 \mathfrak{M}^n , 30
 \mathbb{N} , 9
 \mathbb{P} , 9
 \mathbb{T} , 10
 \mathbb{T}_n , 11
 \mathbf{V} , 14
 \Vdash , 20
 \aleph_0 , 9
 $\tilde{\eta}^\tau$, 44
 \cdot , 10
 \top , 10
 \bowtie , 40
 $\check{\tau}$, 14
 $\text{dom}(\tau)$, 13
 $\tau[\sigma]$, 13
 \perp , 10
 $\text{int}(\tau, \mathcal{C})$, 15
 \times , 39
 $\tilde{\cdot}$, 14
 $\max(P)$, 42
 $\max(p, q)$, 10
 \oslash , 38, 40
 ϵ , 23, 35
 ϵ_p , 22, 34
 \approx , 24, 38
 \approx_p , 24, 37
 $\min(p, q)$, 10
 $|x|$, 9
 \asymp , 46
 \sqsubseteq , 26, 37
 \sqsubseteq_p , 26, 36
 $\not\sqsubset$, 20
 \neq , 20
 \notin , 23
 \notin_p , 23
 $\not\approx$, 24
 $\not\approx_p$, 24
 \notin , 20
 $\not\subset$, 20
 $p \cdot 0$, 10
 $p \cdot 1$, 10
 $\text{ran}(\tau)$, 13
 P^\top , 41
 σ^p , 13
 P^\perp , 41
 $\tau_{\mathcal{C}}$, 15
 \vdash , 31
 ${}^n 2$, 9
 $\leq^{\mathbb{N}2}$, 9

Indeks pojęć

- ZFC, 12
 - antyłańcuch, 11
 - maksymalny, 11, 31
 - poniżej p , 11
 - baza dopełnienia, 44
 - brzeg metazbioru, 44
 - ciało metazbiorów, 50
 - część wspólna metazbiorów, 40
 - drzewo binarne, 10
 - dziedzina metazbioru, 13
 - element
 - maksymalny, 9
 - minimalny, 9
 - najmniejszy, 9
 - największy, 9
 - potencjalny, 12
 - forsing, 7, 12
 - gałąź, 11
 - interpretacja
 - metazbioru, 15
 - zdania, 20
 - izomorfizm kanoniczny, 14
 - korzeń, 10
 - łańcuch, 11
 - liczba porządkowa, 9
 - metapodzbior, 26
 - metazbiory
 - równe, 24, 38
 - pod warunkiem p , 24, 37
 - różne, 24
 - metazbiór, 12
 - pod warunkiem p , 24
 - n -tego poziomu, 30
 - dziedzicznie skończony, 15
 - kanoniczny, 14
 - pierwszego poziomu, 30
- należenie
 - metazbiorów, 22, 23, 35
 - warunkowe, 22, 23, 34
 - obcięcie, 33
 - obraz elementu potencjalnego, 13
 - pokrywanie, 31
 - porządek
 - częściowy, 9
 - poziomu, 58
 - poziom w drzewie, 11
 - przeciwdziedzina metazbioru, 13
 - półsplot, 39
 - rozpinanie, 31
 - równoważność zbiorów, 32
 - poniżej p , 33
 - równość
 - przybliżona, 24
 - warunkowa, 24, 37
 - różnica metazbiorów, 46
 - splot, 40
 - suma metazbiorów, 38
 - warunek, 10
 - silniejszy, 10
 - słabszy, 10
 - wymuszanie, 20
 - zawieranie
 - metazbiorów, 26, 36, 37
 - warunkowe, 26, 36

zbiór rozmyty
 α -przekrój, 58
 intuicjonistyczny, 60
 jądro, 59
 nośnik, 59

Bibliografia

- [1] Alfred Aho, Ravi Sethi, Jeffrey Ullman. *Kompilatory. Reguły, metody i narzędzia*. Klasyka Informatyki. WNT, 2002.
- [2] Krassimir Todorov Atanassov. Intuitionistic fuzzy sets. *Fuzzy Sets and Systems*, 20:87–96, 1986.
- [3] Krassimir Todorov Atanassov. *Intuitionistic Fuzzy Sets: Theory and Applications*. Studies in Fuzziness and Soft Computing. Physica-Verlag, Heidelberg, 1999.
- [4] Krzysztof Barteczko, Wojciech Drabik, Bartłomiej Starosta. *Programowanie w języku Java. Ćwiczenia z . . .* MIKOM, 2003.
- [5] Krzysztof Barteczko, Wojciech Drabik, Bartłomiej Starosta. *Nowe Metody Programowania, Tom 1*. PJWSTK, 2004.
- [6] Krzysztof Barteczko, Wojciech Drabik, Bartłomiej Starosta. *Nowe Metody Programowania, Tom 2*. PJWSTK, 2006.
- [7] Leonard Bolc, Piotr Borowik. *Many-valued Logics 1: Theoretical Foundations*. Springer, 1992.
- [8] Leonard Bolc, Piotr Borowik. *Many-valued Logics 2: Automated Reasoning And Practical Applications*. Springer, 2004.
- [9] Paul Cohen. The independence of the continuum hypothesis 1. W *Proceedings of the National Academy of Sciences of the United States of America*, tom 50, strony 1143–1148, 12 1963.
- [10] Paul Cohen. The independence of the continuum hypothesis 2. W *Proceedings of the National Academy of Sciences of the United States of America*, tom 51, strony 105–110, 1 1964.
- [11] Paul Cohen. *Set Theory and the Continuum Hypothesis*. Addison-Wesley Longman, 1966.
- [12] Didier Dubois, Henri Prade. Rough fuzzy sets and fuzzy rough sets. *International Journal of General Systems*, 17:191–209, 1990.
- [13] Robin Giles. Łukasiewicz logic and fuzzy set theory. *International Journal of Man-Machine Studies*, 8:313–327, 1976.
- [14] Joseph Goguen. L-fuzzy sets. *Journal of Mathematical Analysis and Applications*, 18:145–174, 1967.

- [15] Witold Kosinski. Calculation and reasoning with ordered fuzzy numbers. In *EUSFLAT Conf.*, pages 412–417, 2005.
- [16] Witold Kosiński, Piotr Prokopowicz. Algebra liczb rozmytych. *Matematyka Stosowana*, (5), 2004.
- [17] Kenneth Kunen. *Set Theory, An Introduction to Independence Proofs*. Numer 102 w *Studies in Logic and Foundations of Mathematics*. North-Holland Publishing Company, 1980.
- [18] Kazimierz Kuratowski, Andrzej Mostowski. *Teoria mnogości*. Polskie Towarzystwo Matematyczne, 1952.
- [19] Jan Łukasiewicz. O logice trójwartościowej. *Ruch filozoficzny*, 5:170–171, 1920.
- [20] Jan Łukasiewicz. On three-valued logic. W Ludwik Borkowski, redaktor, *Selected works by Jan Łukasiewicz*, *Studies in logic and the foundations of mathematics*, strony 87–88. North-Holland Publishing Company, 1970.
- [21] Storrs McCall, editor. *Polish Logic 1920-1939*. Oxford University Press US, 1967.
- [22] Vilém Novák. Fuzzy sets – the approximation of semisets. *Fuzzy Sets and Systems*, 14:259–272, 1984.
- [23] Zdzisław Pawlak. Rough sets. *International Journal of Computer and Information Sciences*, 11:341–356, 1982.
- [24] Andrzej Piegat. *Modelowanie i sterowanie rozmyte*. *Problemy Współczesnej Nauki. Teoria i Zastosowania*. Akademicka Oficyna Wydawnicza EXIT, 1999.
- [25] Helena Rasiowa. *Wstęp do matematyki współczesnej*. Biblioteka Matematyczna, Tom 30. PWN, 1990.
- [26] Bartłomiej Starosta. Cardinality and equipotency of metasets. W przygotowaniu.
- [27] Bartłomiej Starosta. *JMX. Zarządzanie aplikacjami w języku Java*. PJWSTK, 2006.
- [28] Bartłomiej Starosta. Application of meta sets to character recognition. *Proceedings of the 18th International Symposium on Methodologies for Intelligent Systems*, *Lecture Notes in Artificial Intelligence*, Springer 2009. W druku.
- [29] Bartłomiej Starosta, Witold Kosiński. Forcing relation for hereditarily finite meta sets. W przygotowaniu.
- [30] Bartłomiej Starosta, Witold Kosiński. *Intuitionistic Fuzzy Sets: Recent Advances*, rozdział Meta Sets. A New Concept of Intuitionistic Fuzzy Sets. *Studies in Fuzziness and Soft Computing*. Springer Verlag, 2009.

-
- [31] Bartłomiej Starosta, Witold Kosiński. *Views on Fuzzy Sets and Systems from Different Perspectives. Philosophy and Logic, Criticisms and Applications*, tom 243 w *Studies w Fuzziness and Soft Computing*, rozdział Meta Sets. Another Approach to Fuzziness, strony 509–522. Springer Verlag, 2009.
- [32] Petr Vopěnka, Petr Hájek. *The Theory of Semisets*. North-Holland Publishing Company, 1972.
- [33] Maciej Wygralak. *Cardinalities Of Fuzzy Sets*. Springer Verlag, 2003.
- [34] Lotfi Asker Zadeh. Fuzzy sets. *Information and Control*, 8:338–353, 1965.
- [35] Lotfi Asker Zadeh. Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Transactions on Systems, Man and Cybernetics*, 3:28–44, 1973.
- [36] Lotfi Asker Zadeh. The concept of a linguistic variable and its application to approximate reasoning. *Information Sciences*, 8:199–249, 1975.
- [37] Lotfi Asker Zadeh. The birth and evolution of fuzzy logic. *International Journal of General Systems*, 17:95–105, 1990.
- [38] Lotfi Asker Zadeh. Fuzzy logic = computing with words. *IEEE Transactions on Fuzzy Systems*, 4(2):103–111, 1996.