

Autoreferat

Filip Mazowiecki

28 listopada 2025

1 Imię i nazwisko

Filip Mazowiecki

2 Dyplomy

Stopnie naukowe

- **Luty 2016** doktor nauk w dziedzinie informatyki, Uniwersytet Warszawski, tytuł rozprawy: *Frontiers of tractability for recursive queries*, promotorzy: Filip Murlak i Emanuel Kieroński;
- **Marzec 2013** magister informatyki, Uniwersytet Warszawski, tytuł pracy: *Alpha-Conversion on generalised nominal sets*, promotor: Bartek Klin;
- **Czerwiec 2011** magister matematyki, Uniwersytet Warszawski, tytuł pracy: *Continuous reductions of regular languages*, promotor: Filip Murlak.

3 Zatrudnienie

- **Od lutego 2022** Adiunkt im. Samuela Eilenberga (assistant professor) na Uniwersytecie Warszawskim;
- **Wrzesień 2019 – Styczeń 2022** research group leader w Max Planck Institute for Software Systems;
- **Grudzień 2017 – Sierpień 2019** postdoc na Uniwersytecie w Bordeaux;
- **Kwiecień 2017 – Listopad 2017** postdoc na Uniwersytecie Oksfordzkim;
- **Marzec 2016 – Marzec 2017** postdoc na University of Warwick.

4 Opis osiągnięć

Tytuł: “Weighted automata: expressiveness and decidability”.

4.1 Tło

Automaty ważone Automaty ważone są naturalnym modelem obliczeniowym, który zamiast rozpoznawać słowa wejściowe przypisuje im wartości liczbowe. Tradycyjne modele przypisują każdemu słowu jedną z dwóch wartości: 0 (odrzuć) lub 1 (akceptacja). Automaty ważone przypisują wartości z szerszych dziedzin; najbardziej intuicyjny wariant — *automaty probabilistyczne* — przypisuje prawdopodobieństwo akceptacji danego słowa. W ogólności dziedzina może być dowolnym przemennym półpierścieniem. Wówczas automaty skończone można traktować jako automaty ważne nad półpierścieniem boolowskim.

W niniejszym dokumencie interesują nas dwie klasy półpierścieni. Po pierwsze, ciała, a dokładniej ciała liczb wymiernych ze standardowym dodawaniem i mnożeniem; można je dodatkowo ograniczyć do liczb wymiernych z przedziału $[0, 1]$ przy automatach probabilistycznych. Po drugie, półpierścienie tropikalne, gdzie dziedzina jest zazwyczaj ograniczona do liczb całkowitych lub naturalnych, operacja dodawania w półpierścieniu to $\max$ lub $\min$, a mnożenie to $+$.

Dla obu klas półpierścieni badane problemy są często podobne, a czasem wręcz te same, lecz stosowane techniki i teoria zazwyczaj się różnią. Nie istnieje jedno ujednolicone podejście i wydaje się mało prawdopodobne, by takie mogło istnieć. Ten sam problem może być rozstrzygalny nad jednym półpierścieniem i nierozstrzygalny nad innym. Na przykład równoważność automatów ważonych jest rozstrzygalna w czasie wielomianowym nad ciałami [75], ale jest nierozstrzygalna nad półpierścieniami tropikalnymi [2].

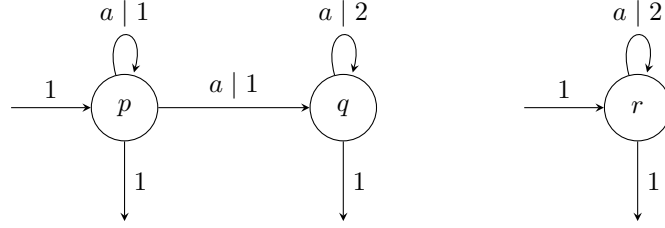
Ustalmy przemienność półpierścień $\mathbb{S}(\oplus, \odot)$. Istnieją dwie naturalne definicje automatów ważonych nad $\mathbb{S}(\oplus, \odot)$, które są równoważne, ale dają różną intuicję. Zaczniemy od definicji podkreślającej interpretację jako przekształcenia liniowe. Automat ważony to krotka $\mathcal{A} = (d, \Sigma, (M_a)_{a \in \Sigma}, I, F)$, gdzie: $d \in \mathbb{N}$ to wymiar; Σ to skończony alfabet; każda $M_a \in \mathbb{S}^{d \times d}$ jest macierzą przejścia; a $I, F \in \mathbb{S}^d$ są odpowiednio wektorem początkowym i końcowym. Dla słowa $a_1 \dots a_n \in \Sigma^*$ automat zwraca wartość

$$I^\top \odot M_{a_1} \odot \dots \odot M_{a_n} \odot F.$$

Mnożenie macierzy jest dobrze określone nad przemennymi półpierścieniami. Z tego punktu widzenia łatwo zauważyć, że automaty ważone uogólniają liniowe ciągi rekurencyjne (LRS). Rzeczywiście, gdy alfabet ma jedną literę, każde słowo utożsamiamy z jego długością, tzn. $\Sigma^* \equiv \mathbb{N}$. Zatem automaty ważone nad alfabetem jednoelementowym definiują naturalnie n -ty wyraz ciągu. Gdy $\mathbb{S}(\oplus, \odot)$ jest ciałem liczb wymiernych, dostajemy standardową klasę LRS, gdzie typowym przykładem jest ciąg Fibonacciego, patrz Fig. 1 oraz Section 4.2.

Druga definicja jest równoważna, niemal syntaktycznie, lecz używa innych nazw: $\mathcal{A} = (Q, \Sigma, (M_a)_{a \in \Sigma}, I, F)$. Wymiar d zastępujemy skończonym zbiorem stanów Q , a wymiary macierzy i wektorów to $|Q|$. Każda macierz M_a jest interpretowana jako ważona macierz przejść między stanami. Wektory I i F przechodzą wagi początkowe i końcowe stanów. Wyjście automatu definiujemy wtedy tak: każdy przebieg nad słowem wnosi do wyniku iloczyn ($\odot$) wszystkich wag przejść oraz wag stanu początkowego i końcowego, a wartość automatu jest sumą ($\oplus$) po wszystkich biegach. Dzięki prawu rozdzielności w przemennych półpierścieniami definicja ta jest równoważna z pierwszą, patrz Fig. 1. Łatwo też zauważyć, że nad półpierścieniem boolowskim automaty ważne są równoważne automatom skończonym w tym sensie, że akceptowane słowa to te, którym przypisano wartość prawdy.

Drugi punkt widzenia pozwala w naturalny sposób wprowadzać ograniczenia znane z teorii automatów. Przykładem jest ograniczenie *unambiguous*. Intuicyjnie są to automaty ważne definiujące funkcje, które przy obliczaniu wyniku nie używają $\oplus$. W pierwszej definicji trudno ująć to zgrabnie formalnie, natomiast w drugiej wystarczy powiedzieć, że dla każdego słowa istnieje co najwyżej jeden bieg o niezerowej wadze. Więcej szczegółów poniżej.



Rysunek 1: Przykłady automatów ważonych nad alfabetem jednoelementowym $\{a\}$. Rysunki odpowiadają drugiej definicji, gdzie przejścia o wadze 0 i stany z wagą początkową/końcową 0 są pominięte. Można je przekształcić do pierwszej definicji. Po prawej: wymiar $d = 1$, jedyna macierz $M_a = (2)$, $I = F = (1)$; automat przypisuje $a^n \mapsto 2^n$. Po lewej: wymiar $d = 2$, jedyna macierz $M_a = \begin{pmatrix} 1 & 1 \\ 0 & 2 \end{pmatrix}$ (stan p odpowiada pierwszej współrzędnej, a q drugiej), $I^\top = (1, 0)$ i $F^\top = (1, 1)$. Łatwo sprawdzić przez indukcję, że po przeczytaniu a^n mamy $I^\top(M_a)^n = (1, 2^n - 1)$, zatem automat po lewej przypisuje $a^n \mapsto 1 + 2^n - 1 = 2^n$.

Ambiguity Istnieje kilka podejść do definiowania podklas automatów ważonych. Jedno z nich polega na ograniczaniu liczby przebiegów akceptujących. Dla danego $\mathcal{A}$ wyróżniamy cztery klasy:

- automaty unambiguous, tzn. z co najwyżej jednym biegiem akceptującym;
- automaty finitely-ambiguous, tzn. liczba biegów akceptujących jest ograniczona pewną stałą niezależną od długości słowa;
- automaty polynomially-ambiguous, tzn. liczba biegów jest ograniczona wielomianem od długości słowa;
- pełna klasa, nazywana w tym kontekście czasem nazywana exponentially-ambiguous.

Zauważmy, że definicje tych klas zależą wyłącznie od struktury automatu; z punktu widzenia klasy istotne jest jedynie to, czy wagi są zerowe czy nie. Przynależność automatu do jednej z klas jest bardzo dobrze zrozumiana i równoważna zakazaniu pewnych prostych wzorców [78]. Na podstawie tych charakterystyk można wnioskować też, że są to „najważniejsze” klasy: klasy pośrednie stają się równoważne jednej z nich. Przykładowo klasa „logarithmic-ambiguous” pokrywa się z finitely-ambiguous.

Jeżeli automat nie należy do danej klasy, nie znaczy to, że nie istnieje automat równoważny należący do tej klasy. Tak jest w Fig. 1, gdzie automat po lewej jest polynomially ambiguous, natomiast równoważny mu automat po prawej jest unambiguous. Pojawia się więc naturalne pytanie, czy ta hierarchia klas funkcji jest ścisła. Odpowiedź zależy od modelu automatów, a dokładniej — od rozważanego półpierścienia. Przykładowo, dla automatów skończonych cała hierarchia zapada się dzięki konstrukcji zbioru potęgowego (powerset construction). Dla półpierścieni tropikalnych znane są przykłady rozdzielające klasy: istnieje automat finitely-ambiguous, którego nie można zrealizować automatami unambiguous [53, Section 3.4], oraz automat polynomially-ambiguous, którego nie można zrealizować automatami finitely-ambiguous [52, Proposition 3.2].

Słabością tych prac jest brak uogólnionej metody dowodzenia wyników separacyjnych dla innych funkcji realizowanych przez automaty ważne. Co więcej, nie istniał formalny dowód oddzielający klasę polynomially-ambiguous od klasy exponentially-ambiguous.

Klasy niejednoznaczności dla automatów ważonych nad ciałami zostały po raz pierwszy szczegółowo zbadane w 2017 roku [44], konkretniej w kontekście automatów probabilistycznych. Autorzy skupili się tam na problemach decyzyjnych, nie podając przykładów rozdzielających klasy.

$$a_0 = 2, \quad a_{n+1} = (a_n)^2 \qquad \qquad \begin{cases} b_0 = 1 \\ c_0 = 1 \end{cases}, \quad \begin{cases} b_{n+1} = b_n \cdot c_n \\ c_{n+1} = c_n + 1. \end{cases}$$

Rysunek 2: Po lewej ciąg $a_n = 2^{2^n}$ zdefiniowany przez wielomian x^2 . Po prawej ciąg $b_n = n!$, definiowany przy pomocy pomocniczego ciągu $c_n = n + 1$ i układu wielomianów xy i $y + 1$. Liczba zmiennych odpowiada liczbie ciągów.

Cost-register automata i ograniczenie copyless Cost-register automata (CRA) [5], znane też jako polynomial automata [14] lub polynomial recurrent relations [76], uogólniają automaty ważone poprzez dopuszczenie nieliniowych aktualizacji. Pomińmy formalną definicję; wystarczy myśleć, że aktualizacje liniowe zostają zastąpione aktualizacjami zadanymi przez wielomiany. W ten sposób automaty ważone uogólniają liniowe ciągi rekurencyjne, a cost-register automata — ciągi rekurencyjne o wielomianowej postaci, patrz Fig. 2.

Będziemy używać nazwy CRA, gdyż w tym kontekście wygodnie zdefiniować ograniczenie *copyless*, omawiane szczegółowo w dalszej części. Model CRA i ograniczenie copyless są często analizowane w kontekście modeli transduktorów [71]. Ograniczenie copyless dotyczy przejść aktualizujących: intuicyjnie, przy jednej aktualizacji nie można użyć wartości tej samej zmiennej więcej niż raz. W przykładach z Fig. 2 oba ciągi a_n i b_n naruszają to ograniczenie. Dla a_n używamy wielomianu x^2 , który formalnie jest iloczynem $x \cdot x$, co wykorzystuje wartość x dwukrotnie. Dla b_n zmienna y pojawia się zarówno w xy , jak i w $y + 1$; bardziej intuicyjnie, wartość c_n jest potrzebna dwa razy. Prosty przykład automatu copyless to sam ciąg $c_n = n + 1$, gdyż definiuje go wielomian $y + 1$. Uogólniając ten przykład, można pokazać, że dowolny ciąg wielomianowy da się zdefiniować przy użyciu copyless CRA.

Przy takim ujęciu CRA, łatwo zauważyć, że ograniczenie się do aktualizacji liniowych prowadzi do modelu równoważnego automatom ważonym. Formalny dowód takiej równoważności znajduje się w [5]. Podsumowując, otrzymujemy następujące klasy funkcji:

- pełna klasa CRA;
- copyless CRA;
- liniowe CRA;
- liniowe i copyless CRA.

Pierwsza z nich jest oczywiście najbardziej ekspresywna, a ostatnia — najmniej. Nie jest jednak a priori jasne, jak porównać środkowe dwie klasy, ani jak copyless CRA oraz liniowe copyless CRA mają się do klas definiowanych przez ograniczenia niejednoznaczności, np. automaty polynomially-ambiguous.

Częściowe odpowiedzi pojawiają się w kilku pracach, których jestem współautorem, ale nie zaliczyłem ich do głównych osiągnięć w tym dokumencie. W [66] i [65, Lemma 5] pokazano, że klasa copyless CRA zawiera się w klasie liniowych CRA, czyli równoważnie — w klasie wszystkich automatów ważonych — nad dowolnym przemiennym półpierścieniem. Dowód nie jest trudny: mając copyless CRA, budujemy liniowe CRA, którego rejestrami są wszystkie monomiany, w których każdy rejestr występuje co najwyżej raz. Aktualizacje przejmujemy z oryginalnego CRA; dzięki ograniczeniu copyless, nowe aktualizacje są liniowe.

W [3, 4] pokazaliśmy, że klasa liniowych copyless CRA zawiera się, nad dowolnym półpierścieniem,

w klasie automatów polynomially-ambiguous¹. Uzupełniają to wyniki pokazujące, że liniowe copyless CRA nie obejmują nawet klasy automatów unambiguous, patrz np. Sekcja 4 pracy przeglądowej [38].

Dalsza część dokumentu podzielona jest tematycznie. W Section 4.2 i Section 4.3 omawiam wyniki dotyczące wyrażalności automatów ważonych i ich rozszerzeń nad ciałem liczb wymiernych oraz nad półpierścieniami tropikalnymi. W Section 4.4 omawiam wyniki dotyczące problemów decyzyjnych dla automatów ważonych nad ciałami.

4.2 Wyrażalność nad ciałem liczb wymiernych

Wyniki w tej sekcji dotyczą automatów ważonych i CRA nad ciałem liczb wymiernych, ale ograniczonych do alfabetów jednoliterowych. Przypomnijmy, że dla alfabetu $\Sigma = \{a\}$ słowa mają postać a^n i utożsamiamy je z ich długością. Automat ważony nad liczbami wymiernymi jest więc funkcją $\mathbb{N} \rightarrow \mathbb{Q}$. Definicja macierzowa opisuje ciągi postaci $a_n = I^T M^n F$. Tę klasę nazywamy linear recurrence sequences (LRS). Alternatywną, klasyczną definicją jest pojedyncze równanie rekurencyjne stałej głębokości: $a_{n+k} = \sum_{i=0}^{k-1} c_i \cdot a_{n+i}$. Przykładowo ciąg Fibonacciego spełnia $F_{n+2} = F_{n+1} + F_n$. Obie definicje są efektywnie równoważne patrz np. [23, Proposition 2.1].

Taka perspektywa prowadzi do naturalnych pytań: jakie ciągi można wyrazić przy użyciu ograniczonych wariantów automatów ważonych i CRA? To jest tematem badań w pierwszej z omawianych prac.

- (A) Corentin Barloy, Nathanaël Fijalkow, Nathan Lhote, and Filip Mazowiecki.

A robust class of linear recurrence sequences.

Information and Computation, 2022 [7]; wersja konferencyjna w *CSL*, 2020 [6].

Pod względem technicznym jest to najprostsza praca z tego zestawu i może być traktowana jako łagodne wprowadzenie do typu wyników prezentowanych w pozostałych częściach. Najbardziej intuicyjny wynik to że w tym kontekście klasy ciągów odpowiadające automatom polynomially-ambiguous i copyless CRA są sobie równe. Co więcej, są one równoważne z klasą LRS, których wartości własne są pierwiastkami liczb wymiernych, tzn. liczbami zespolonymi z takimi, że z^n jest wymierne dla pewnego $n \in \mathbb{N}$. Wartości własne są pierwiastkami wielomianu odpowiadającego równaniu rekurencyjnemu; np. ciąg Fibonacciego spełnia $F_{n+2} - F_{n+1} - F_n = 0$, co daje wielomian $x^2 - x - 1 = 0$, a jedną z wartości własnych jest złoty podział φ . Wniosek: automaty polynomially-ambiguous i copyless CRA ściśle mniej wyrażają niż pełna klasa automatów ważonych, gdyż φ^n nigdy nie jest liczbą wymierną. Pozostałe wyniki z pracy (A) charakteryzują klasy LRS odpowiadające innym fragmentom automatów ważonych i CRA.

Choć równoważność polynomially-ambiguous i copyless CRA w tym modelu jest elegancka, nie była w pełni satysfakcjonująca: wierzyliśmy, że dla alfabetów nieograniczonych klasy te różnią się. Niedawno udało nam się to pokazać w nieopublikowanej pracy [65] (nieomawianej tutaj). Warto dodać, że treść (A) jest inspirowana sekcją 4 pracy [69], omawianej w Section 4.3.

Druga praca omawiana w tej sekcji dotyczy rozszerzeń automatów ważonych nad ciałami, rozpatrywanych jako klasy ciągów. Wspominaliśmy już, że CRA można postrzegać jako rozszerzenie LRS poprzez rekurencje zdefiniowane wielomianami; klasę tę nazywamy *polynomially recursive sequences* (Polyrec). W literaturze pojawiają się jednak jeszcze dwa inne naturalne rozszerzenia automatów ważonych. Pierwsze pochodzi z tego że gramatyki bezkontekstowe rozszerzają automaty skończone; można to przenieść w świat automatów ważonych [45]. Odpowiadająca klasa ciągów to dobrze znana klasa *holonomic sequences*, patrz np. książka [51]. Tam rekurencje mogą używać bieżącego indeksu

¹W [3, 4] wszystkie CRA są z założenia liniowe, dlatego model ten nazywany jest tam copyless CRA, co może być mylące w odniesieniu do obecnej notacji.

n po obu stronach równania. Przykładowo $a_n = n!$ można zdefiniować przez $a_{n+1} = n \cdot a_n$, a liczby Catalana spełniają $(n+2) \cdot C_{n+1} = (4n+2)C_n$.

Drugie rozszerzenie pochodzi z logiki. Istniały badania zmierzające do logicznej charakterystyki automatów ważonych, analogicznej do charakterystyki automatów skończonych przez MSO. Ostatecznie otrzymane logiki okazały się bardziej ekspresywne niż automaty ważne [42, 55]. Intuicyjnie kwantyfikatory $\exists$ i $\forall$ zastąpiono operatorami sumy i iloczynu. Odpowiadającej klasie ciągów nadamy roboczo nazwę *MSO sequences*. Przykładami ciągów definiowalnych w tej klasie są n^n oraz $n!$, np. za pomocą formuł:

$$\prod_x \sum_y 1, \quad \prod_x \sum_{y \leq x} 1.$$

Intuicyjnie formuła 1 przyjmuje wartość 1 na każdej pozycji słowa (tu a^n). Wówczas $\sum_y 1$ zwraca długość słowa n , a $\prod_x \sum_y 1$ bierze iloczyn n takich składników, zwracając n^n . W drugiej formule ograniczenie $y \leq x$ sprawia, że mnożymy wszystkie liczby od 1 do n .

Naturalne pytanie brzmi: jak porównać trzy klasy ciągów — Polyrec, holonomic sequences i MSO sequences? Przypuszczamy, że są wzajemnie nieporównywalne. Jedynymi trywialnymi nieinkluzjami są te, że holonomic sequences nie mogą zawierać ani Polyrec, ani MSO sequences ze względu na ograniczenia asymptotyczne: łatwo zauważyć, że holonomic sequences są asymptotycznie ograniczone przez $2^{\text{poly}(n)}$, podczas gdy np. ciąg 2^{2^n} jest Polyrec (por. Fig. 2) i da się go również zdefiniować w klasie MSO sequences. W kolejnej pracy pokazujemy dwie dalsze nieinkluzje.

(B) Michaël Cadilhac, Filip Mazowiecki, Charles Paperman, Michal Pilipczuk i Géraud Sénizergues.

On polynomial recursive sequences.

Theory of Computing Systems (TOCS) 2021 [23]; wersja konferencyjna *ICALP* 2020 [22].

W pracy (B) pokazujemy, że Polyrec nie może zawierać ani holonomic sequences, ani MSO sequences. Osiągamy to, dowodząc, że ani liczby Catalana, ani ciąg n^n nie mogą być zdefiniowane za pomocą Polyrec. Pierwszy wynik jest koncepcyjnie prosty: wystarczy analiza ciągów modulo liczby pierwsze. Okazuje się, że dla Polyrec ciągi modulo liczby pierwszej muszą być ostatecznie okresowe. To wystarczy, ponieważ np. liczba Catalana C_n jest nieparzysta wtedy i tylko wtedy, gdy n ma postać $2^k - 1$.

Drugi wynik jest koncepcyjnie znacznie trudniejszy. Najpierw trzeba było znaleźć własność charakterystyczną ciągów Polyrec, której n^n nie spełnia. Wprowadzamy pojęcie *cancelling polynomials*. Dla każdego ciągu Polyrec istnieje niezerowy wielomian wiążący skończoną liczbę kolejnych wartości ciągu. Dla $a_n = n!$ łatwo sprawdzić, że

$$\frac{a_{n+2}}{a_{n+1}} = \frac{a_{n+1}}{a_n} + 1,$$

co prowadzi do zależności

$$a_{n+2}a_n - a_{n+1}^2 - a_{n+1}a_n = 0.$$

Posiadanie cancelling polynomial nie charakteryzuje jednak Polyrec; np. dowolny ciąg b_n o wartościach w $\{-1, 1\}$ (np. nie musi być ostatecznie okresowy) spełnia $b_n^2 - 1 = 0$, co definiuje cancelling polynomial.

Pokazujemy, że ciągi Polyrec zawsze mają cancelling polynomials, a głównym punktem pracy jest dowód, że ciąg n^n nie ma takiej zależności — każda taka zależność prowadziłaby do nienaturalnego powiązania między kolejnymi wyrazami, co okazuje się niemożliwe.

Na koniec warto zauważyć, że cancelling polynomial dla $n!$ nie jest liniowy w a_{n+2} . W pracy (B) prostym argumentem asymptotycznym pokazujemy, że nie istnieje cancelling polynomial liniowy w wyrazie o największym indeksie. W późniejszej pracy proponujemy hipotezę, że liniowość można uzyskać, jeśli dopuścimy funkcje wymierne zamiast wielomianów [30]. Taki wynik miałby konsekwencje dla górnych oszacowań złożoności problemu równoważności dla Polyrec. Obecnie wiadomo, że problem jest PSPACE-trudny i osadzony w klasie Ackermanna [30, Section 5].

4.3 Wyrażalność nad półpierścieniami tropikalnymi

W tej sekcji omawiam wyniki dotyczące wyrażalności automatów ważonych nad półpierścieniami tropikalnymi. Aby uniknąć technicznych subtelności, zakładamy zazwyczaj, że dziedziną są liczby naturalne, dodawanie to $\min$, a mnożenie to $+$. Niekiedy zamiast $\min$ stosuje się $\max$; będzie to zawsze wyraźnie zaznaczone.

Wiadomo było, że w tym kontekście automaty finitely-ambiguous mogą wyrazić więcej niż unambiguous [54], a automaty polynomially-ambiguous — bardziej ekspresywne niż finitely-ambiguous [53]. Wyniki te uzyskano jednak poprzez analizę konkretnych przykładów, bez ogólnej techniki dowodzenia niewyrażalności. To był punkt wyjścia dla kolejnej omawianej pracy.

- (C) Agnishom Chattopadhyay, Filip Mazowiecki, Anca Muscholl i Cristian Riveros (wersja konferencyjna: Filip Mazowiecki i Cristian Riveros).

Pumping lemmas for weighted automata.

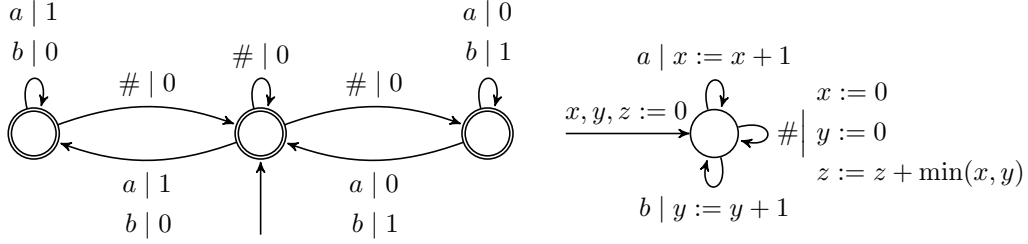
Logical Methods in Computer Science (LMCS) 2021 [25]; wersja konferencyjna *STACS* 2018 [68].

W pracy tej przedstawiamy lematy pompujące dla klas unambiguous, finitely-ambiguous oraz polynomially-ambiguous automatów ważonych nad półpierścieniami tropikalnymi (zarówno z $\min$, jak i z $\max$). Pokazujemy, jak stosować te lematy jako czarne skrzynki, aby uzyskać krótkie dowody separacji klas. Praca zawiera również pierwsze formalne dowody rozdzielenia klasy polynomially-ambiguous od exponentially-ambiguous. Dzięki temu pokazujemy także, że polynomially-ambiguous automata nie zawierają klasy copyless CRA. Funkcję świadkującą te fakty przedstawiono na Fig. 3. W pewnych sytuacjach nasze lematy można stosować także w klasach szerszych niż automaty ważne, gdzie pojawiają się nieliniowe użycia operacji mnożenia (czyli dodawania w odpowiadających półpierścieniach tropikalnych).

Główną ideą przy formułowaniu lematów pompujących było znalezienie właściwego sposobu rozcinania słów na części, które można pompować niezależnie. Intuicyjnie, w klasie finitely-ambiguous powinno istnieć stałe ograniczenie na liczbę fragmentów, które mogą niezależnie wpływać na wynik. Główna trudność polegała na odpowiednim sformułowaniu lematów; po podaniu poprawnego sformułowania dowody są bardziej techniczne niż koncepcyjne. Wyjątkiem są dowody dotyczące klasy polynomially-ambiguous, które okazały się najtrudniejsze.

Warto zauważyć, że dla alfabetów jednoelementowych wiadomo było wcześniej, iż pełna klasa (exponentially-ambiguous) automatów ważonych jest równoważna klasie unambiguous [58].

Ostatnia praca omawiana w tej sekcji dotyczy klasy copyless CRA. W czasie jej powstawania nie było jasne, jak klasa ta porównuje się do klas automatów ważonych definiowanych przez ograniczenia niejednoznaczności. Intuicyjnie wydawała się podobna do polynomially-ambiguous, co zostało potwierdzone w szczególnej sytuacji w pracy (A), lecz w ogólności sytuacja jest bardziej subtelna.



Rysunek 3: Przykład funkcji definiwalnej zarówno przez automaty ważone, jak i copyless CRA, ale nieprzez automaty polynomially-ambiguous. Po lewej automat exponentially-ambiguous, po prawej copyless CRA. Stany początkowe/akceptujące po lewej oznaczają wagę początkową/końcową równą 0 (dla pozostałych stanów jest to $+\infty$, czyli zero półpierścienia). Automat po prawej zwraca $z + \min(x, y)$. Alfabet to $\Sigma = \{a, b, \#\}$; słowa należy czytać jako ciągi $(a+b)^*$ rozdzielone symbolami $\#$. Dla słowa $w = w_0\#w_1\#\dots w_n\#$, gdzie $w_i \in (a+b)^*$, wartość funkcji wynosi $\sum_{i=0}^n \min(|w_i|_a, |w_i|_b)$, gdzie $|w_i|_a, |w_i|_b$ oznaczają liczby liter a i b w w_i . Intuicja za brakiem reprezentacji polynomially-ambiguous to że dla każdego i automat musi zdecydować, czy będzie liczyć a , czy b .

(D) Filip Mazowiecki i Cristian Riveros.

Copyless Cost-Register Automata: Structure, Expressiveness, and Closure Properties.

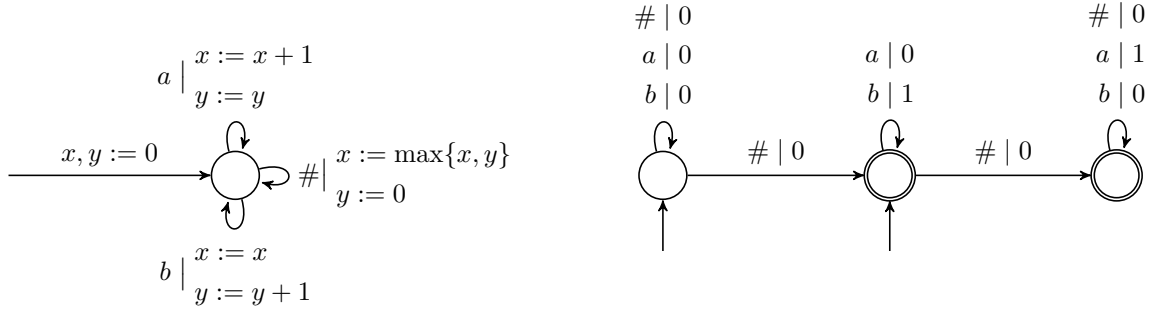
Journal of Computer and System Sciences (JCSS) 2019 [69]; wersja konferencyjna *STACS* 2016 [67].

W pracy (D) dokładnie analizujemy klasę copyless CRA. Artykuł dzieli się zasadniczo na dwie części. W pierwszej pokazujemy różne własności copyless CRA — zarówno w pełnej ogólności nad dowolnym półpierścieniem, jak i w szczególności nad półpierścieniem tropikalnym. W tej części pokazujemy m.in., że istnieje funkcja definiwalna przez automaty polynomially-ambiguous, której nie można zdefiniować za pomocą copyless CRA. W połączeniu z wynikami z (C) pokazuje to, że klasy te są nieporównywalne. Ponadto zauważamy, że copyless CRA nie są domknięte ze względu na odwracanie słowa wejściowego: istnieje copyless CRA $\mathcal{C}$ takie, że nie istnieje copyless CRA $\mathcal{C}'$ z własnością $\mathcal{C}(w) = \mathcal{C}'(w^r)$ dla dowolnego słowa w (gdzie w^r oznacza słowo odwrotne). Przykładową funkcję ilustruje Fig. 4.

W drugiej części pracy analizujemy fragment copyless CRA, który nazywamy *bounded alternation copyless CRA* (BAC). Intuicyjnie „bounded alternation” odnosi się do liczby zmian między sumą i iloczynem w obliczaniu wyniku. Jak można się spodziewać, przykład z Fig. 4 nie należy do BAC: czytanie słów z języka $(a\#)^*$ powoduje nieograniczoną liczbę alternacji między $\max$ a $+$ na rejestrze x . Klasa BAC jest jednak bogatą podklasą copyless CRA, dla której udowadniamy wiele własności domknięcia, m.in. domknięcie ze względu na odwracanie słowa wejściowego. W innej pracy, opartej na (D), pokazaliśmy, że BAC ma ładną charakterystykę logiczną (dla pełnej klasy copyless CRA to wydaje się mało prawdopodobne ze względu na brak odpowiednich własności domknięć) [66].

4.4 Problemy decyzyjne dla automatów ważonych nad ciałami

W tej sekcji zajmujemy się problemami decyzyjnymi wyłącznie dla automatów ważonych nad ciałem liczb wymiernych $\mathbb{Q}(+, \cdot)$. Model ten został wprowadzony już w latach 60. w klasycznej pracy Schützenbergera [75]. Wiele problemów można zredukować do zrozumienia zbiorów osiągalnych obiektów. Dla słów wejściowych $w = a_1 \dots a_n \in \Sigma^*$ rozważamy:



Rysunek 4: Po lewej copyless CRA, po prawej automat polynomially-ambiguous rozpoznający tę samą funkcję. Copyless CRA zwraca $\max(x, y)$. Notacja jest analogiczna jak w Fig. 3, np. stany początkowe i końcowe. Zauważmy, że tutaj suma to $\max$, a nie $\min$. Funkcja jest zdefiniowana następująco: dla słowa $w = w_0\#w_1\#\dots w_n\#$ wartość wynosi $\max_{i=0}^n (|w_i|_b + \sum_{j>i} |w_j|_a)$. Pokazujemy, że po odwróceniu słowa funkcji tej nie da się rozpoznać przez copyless CRA. Ponieważ automaty polynomially-ambiguous są domknięte na odwracanie słowa, przykład ten rozdziela te dwie klasy.

- wartości wyjściowe, tj. zbiór $S_1 = \{I^\top \cdot M_{a_1} \cdot \dots \cdot M_{a_n} \cdot F \mid w \in \Sigma^*\}$;
- wektory konfiguracji, tj. d -wymiarowe wektory $S_2 = \{I^\top \cdot M_{a_1} \cdot \dots \cdot M_{a_n} \mid w \in \Sigma^*\}$;
- macierze przejść, tj. $d \times d$ macierze $S_3 = \{M_{a_1} \cdot \dots \cdot M_{a_n} \mid w \in \Sigma^*\}$.

Wszystkie te zbiory są trudne do opisanie. Dla automatu ważonego oraz elementu s_i problem *przynależności* (membership) pyta, czy $s_i \in S_i$. Wiadomo, że wszystkie trzy wersje są nierozstrzygalne [72, 73, 44]².

Skupiamy się na problemie $s_1 \in S_1$, który można przeformułować następująco: dany jest automat ważony $\mathcal{A}$ oraz liczba wymierna s_1 ; czy istnieje słowo w takie, że $\mathcal{A}(w) = s_1$? Gdy zastąpimy równość nierównością $\mathcal{A}(w) \leq s_1$, otrzymujemy problem *emptiness* lub *threshold*. Nazwa „emptiness” pochodzi stąd, że problem naturalnie uogólnia klasyczny problem pustki języka akceptowanego przez automat skończony. Uogólnieniem jest problem *containment*, gdzie dla dwóch automatów $\mathcal{A}, \mathcal{B}$ pytamy, czy dla każdego słowa w zachodzi $\mathcal{A}(w) \leq \mathcal{B}(w)$. Wszystkie te problemy są w ogólności nierozstrzygalne. Co ciekawe, problem równoważności, tj. czy $\mathcal{A}(w) = \mathcal{B}(w)$ dla wszystkich w , jest rozstrzygalny w czasie wielomianowym [75].

W 2017 roku wyniki o nierozstrzygalności zostały doprecyzowane w kontekście automatów o ograniczonej niejednoznaczności [44]. Autorzy skupili się na automatach probabilistycznych, gdzie dziedzina to przedział $[0, 1]$, a macierze są stochastyczne. Pokazali oni, że emptiness jest nierozstrzygalny już dla automatów polynomially-ambiguous (a ściślej — quadratically-ambiguous, tzn. gdy liczba biegów jest ograniczona wielomianem kwadratowym), ale staje się rozstrzygalny dla finitely-ambiguous automata probabilistycznych. Pierwsza praca z tej sekcji jest naturalnym rozwinięciem tych wyników.

²Nie znaleźliśmy w literaturze osobnego odniesienia dla wersji $s_2 \in S_2$, ale dowody dla $i = 1, 3$ łatwo adaptować.

- (E) Laure Daviaud, Marcin Jurdzinski, Ranko Lazic, Filip Mazowiecki, Guillermo A. Pérez i James Worrell.

When are emptiness and containment decidable for probabilistic automata?

Journal of Computer and System Sciences (JCSS) 2021 [40]; wersja konferencyjna *ICALP* 2018 [39].

Na początku wzmacniamy wynik o nierozstrzygalności emptiness, pokazując, że problem ten jest nierozstrzygalny już dla automatów linearly-ambiguous. Następnie koncentrujemy się na problemie containment dla finitely-ambiguous probabilistic automata. Identyfikujemy ograniczenie, przy którym problem staje się rozstrzygalny, przy założeniu prawdziwości hipotezy Schanuela.

Ograniczenie to jest dość techniczne; wygodniej jest opisać je poprzez równoważny problem. Zamiast automatów probabilistycznych można rozważać automaty ważone nad liczbami wymiernymi nieujemnymi $\mathbb{Q}_{\geq 0}$. Nasze ograniczenie na problem containment jest równoważne rozważaniu problemu emptiness nad $\mathbb{Q}_{\geq 0}$ (bez wymogu stochastyczności macierzy, ale z założeniem nieujemności wag). Szczegóły omawiamy w Sekcji 3 kolejnej pracy (F). Przypomnijmy, że dla finitely-ambiguous automatów probabilistycznych rozstrzygalność emptiness była znana i dość prosta do wykazania [44]. Zastąpienie przedziału $[0, 1]$ pełnym $\mathbb{Q}_{\geq 0}$ dramatycznie zwiększa trudność problemu. Rozstrzygalność containment dla nieograniczonych finitely-ambiguous automatów probabilistycznych (lub automatów ważonych nad $\mathbb{Q}_{\geq 0}$) pozostaje otwarta.

Przejdźmy do omówienia wyników i roli hipotezy Schanuela. W części o nierozstrzygalności stosujemy standardowe techniki używane przy automatach ważonych [2], gdzie wiele przebiegów „zgaduje” różne błędy w kodowaniu maszyn z licznikami. Trudność polega na takiej analizie liczby przebiegów, aby pozostała liniowa.

Dla finitely-ambiguous główna idea polega na przepisaniu problemu do postaci układu nierówności wykładniczych: skończonej sumy wyrażeń postaci

$$p_i \cdot \prod_{x_j} q_{i,j}^{x_j},$$

gdzie $p_i, q_{i,j} \in \mathbb{Q}_{\geq 0}$ są stałymi, a x_j są zmiennymi naturalnymi. Indeksy i odpowiadają różnym biegom (których jest stała liczba), a $q_{i,j}$ są wagami prostych pętli w biegu, które występują x_j razy. Wynika to z klasycznej dekompozycji na proste cykle, znanej już w pracy Rackoffa o ograniczoności dla VASS [74]. Dla intuicji, rozważmy przykład z pracy (E):

Czy istnieją $x, y \in \mathbb{N}$ takie, że dla danego $p \in \mathbb{Q} \cap [0, 1]$ zachodzi

$$p \left(\frac{1}{2}\right)^x 3^y + (1-p) 2^x \left(\frac{1}{3}\right)^y < 1,$$

co równoważnie można zapisać przy użyciu funkcji wykładniczej:

$$\exp(\log(p) - x \log 2 + y \log 3) + \exp(\log(1-p) + x \log 2 - y \log 3) < 1.$$

Następnie rozważamy zbiór

$$V = \{(x, y) \in \mathbb{R}^2 : e^x + e^y < 1\}$$

oraz punkt $b = (\log(p), \log(1-p))$. Definiujemy też wektory

$$\mathbf{u} = (-\log 2, \log 2), \quad \mathbf{v} = (\log 3, -\log 3),$$

i pytamy: czy istnieją $x, y \in \mathbb{N}$ takie, że

$$b + xu + yv \in V?$$

przedstawiamy to obrazowo w Fig. 5. Odpowiedź brzmi: tak wtedy i tylko wtedy, gdy $p \neq \frac{1}{2}$.

Rzeczywiście, niech

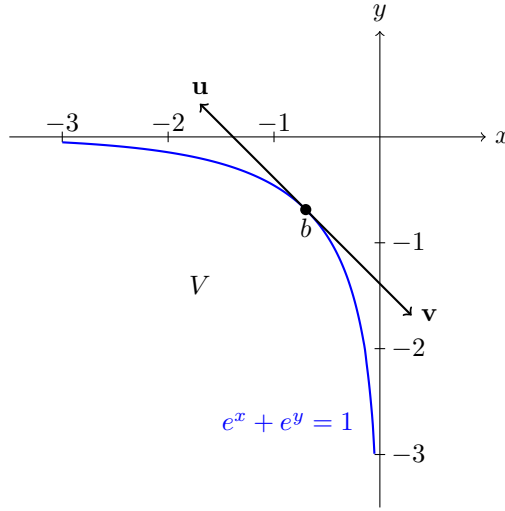
$$C = \{(x, -x) : x \in \mathbb{R}\}.$$

Dla $p = \frac{1}{2}$ prosta afiniczna $C + p$ jest styczna do niebieskiego wykresu, a zatem $b + xu + yv \notin V$ dla wszystkich x, y . Dla $p \neq \frac{1}{2}$ istnieje δ takie, że cały odcinek pomiędzy b a $b + (\delta, -\delta)$ zawiera się w V .

Ponieważ $\log(2)$ i $\log(3)$ są niezależne nad liczbami wymiernymi, zbiór

$$D = \{xu + yv : x, y \in \mathbb{N}\}$$

jest gęsty w C , a więc pewien punkt zbioru $D + b$ leży pomiędzy b a $b + (\delta, -\delta)$, a zatem $b + xu + yv \in V$ dla pewnych $x, y \in \mathbb{N}$.



Rysunek 5: The set V is bounded by the plot $e^x + e^y = 1$ and the point b lies on that plot.

Aby poradzić sobie z tym problemem, wyrażamy go w logice nad liczbami rzeczywistymi z funkcją wykładniczą. Wiadomo, że logika pierwszego rzędu nad $\mathbb{R}(+, \cdot, \exp)$ jest rozstrzygalna, pod warunkiem prawdziwości hipotezy Schanuela [60], która jest matematyczną hipotezą z teorii liczb transcendentalnych.

Wersja czasopismowa pracy zawiera dodatkowo wynik dotyczący wariantu typu promise problem dla emptiness, tzw. *gap emptiness problem*. Dany jest automat probabilistyczny $\mathcal{A}$ i liczba $\varepsilon \in \mathbb{Q} \cap (0, 1)$, przy założeniu, że:

- albo istnieje słowo w takie, że $\mathcal{A}(w) > \frac{1}{2} + \varepsilon$,
- albo dla wszystkich słów w mamy $\mathcal{A}(w) \leq \frac{1}{2}$.

Zadanie polega na rozstrzygnięciu, czy zachodzi druga alternatywa. W ogólności problem ten jest nierozstrzygalny [33], lecz pokazujemy, że jest rozstrzygalny dla polynomially-ambiguous probabilistic automata. Idea algorytmu: (1) wiemy, że problem jest rozstrzygalny dla finitely-ambiguous; (2)

można aproksymować automat polynomially-ambiguous automatem finitely-ambiguous w kontrolowany sposób.

Wynik ten był zaskakujący, ponieważ zwykle problemy nierozstrzygalne dla automatów ważonych pozostają nierozstrzygalne już dla ograniczenia polynomially-ambiguous. Stał się on motywacją dla badania innych problemów, gdzie ograniczenie polynomially-ambiguous mogłoby spowodować że problem staje się rozstrzygalny. Kolejna praca (F) jest bezpośrednim rozwinięciem (E) oraz innej pracy o aproksymacjach automatów ważonych [28].

- (F) Wojciech Czerwinski, Engel Lefauchaux, Filip Mazowiecki, David Purser i Markus A. Whiteland.

The boundedness and zero isolation problems for weighted automata over nonnegative rationals.

Wersja konferencyjna *LICS* 2022 [37]

W pracy (F) badamy dwa problemy dla automatów ważonych nad $\mathbb{Q}_{\geq 0}$: *boundedness* oraz *zero isolation*. Dany automat $\mathcal{A}$ i słowa $w_1, w_2, \dots$; pytamy, czy istnieje ciąg słów taki, że wartości $\mathcal{A}(w_i)$ dążą odpowiednio do $+\infty$ oraz do 0. Oba problemy są w ogólności nierozstrzygalne, a jednocześnie przypuszczamy, że stają się rozstrzygalne dla automatów polynomially-ambiguous. Na pierwszy rzut oka problemy wydają się podobne i można by oczekiwać redukcji między nimi; okazuje się jednak, że wymagają różnych podejść.

Zaczynamy od *boundedness*. Pokazujemy, że problem ten jest rozstrzygalny dla liniowych copyless CRA, czyli klasy zawartej w polynomially-ambiguous [4]³. Idea dowodu polega na analizie półgrupy macierzy S_3 wprowadzonych na początku tej sekcji. W ogólności S_3 jest nieskończony. Możemy przyjąć, że wagi pętli są ograniczone przez 1 — inaczej problem jest trywialny. Następnie obcinamy wartości stosownie małe, co prowadzi do skończonej półgrupy. Dla niej analizujemy szczegółowo elementy idempotentne. Dzięki twierdzeniu Simona o lasach faktoryzacji (Simon's Factorisation Forest Theorem) [77] narzucamy strukturę na świadkach nieograniczoności i redukujemy problem do wykrywania prostych wzorców.

Dla *zero isolation* podejście jest odmienne. Rozważamy z pozoru bardzo prosty fragment polynomially-ambiguous automatów ważonych, nazwany *Independent-CRA*. Nie definiujemy go tu formalnie, gdyż częścią wkładu jest przeformułowanie problemu w bardziej intuicyjnym modelu. Wprowadzamy uogólnienie znanego modelu vector addition systems (VAS) o nazwie *orthant-VASS* (OVAS). Klasyczny VAS w wymiarze d składa się ze skończonego zbioru wektorów z $\mathbb{Z}^d$. W OVAS dopuszczamy wektory z $\mathbb{Q}^{d^+}$ i dodatkowo dla każdego ortantu przestrzeni mamy własny skończony zbiór wektorów.

W problemach dla VAS zazwyczaj mamy dany wektor początkowy i możemy kolejno dodawać wektory z danego zbioru, pod warunkiem, że żadna współrzędna nie spada poniżej zera. Można myśleć, że klasyczny VAS odpowiada OVAS, gdzie tylko dodatni ortant ma niepusty zbiór wektorów. Typowe problemy decyzyjne pytają o osiągalność danej konfiguracji. Łatwo się spodziewać, że większość problemów dla OVAS jest nierozstrzygalna, ponieważ łatwo symulować maszyny z licznikami. My identyfikujemy problem otwarty, którego rozstrzygalność jest równoważna rozstrzygalności *zero isolation* dla ograniczonego modelu *Independent-CRA*: tzw. *universal coverability*. Dla danego OVAS pytamy, czy z każdej konfiguracji początkowej da się dojść do ortantu dodatniego.

Nie udało nam się rozwiązać tego zagadnienia w pełnej ogólności. Pokazaliśmy jednak, że *universal coverability* jest rozstrzygalny w wymiarze 3^5 .

³Wspólnie z Ismaëlem Jeckerem wierzymy, że konstrukcję można rozszerzyć na pełną klasę polynomially-ambiguous; praca ta jest przygotowywana do publikacji i nie jest omawiana w tym dokumencie.

⁴Ścisłej liczby postaci $\log r$, gdzie r jest wymierne; nie wydaje się to istotną przeszkodą i pomijamy szczegóły.

⁵W przypadku ogólnym (przy dopuszczeniu liczb postaci $\log r$) konieczne jest dodatkowe założenie hipotezy Schanuela.

Ostatnia praca dotyczy problemu determinisation. Można go zdefiniować nad dowolnym półpierścieniem: czy dla danego automatu ważonego istnieje automat deterministyczny realizujący identyczną funkcję [59]? Problem ten badano przede wszystkim nad półpierścieniami tropikalnymi; dopiero niedawno pojawił się preprint dający pełne rozwiązanie [1]. Dla ciał rozstrzygalność została pokazana w [9], a następnie szczegółowo przeanalizowana w [10]; górne ograniczenie 2-EXPTIME uzyskano w [13]. Bazując na [9], w pracy (G) analizujemy problem determinisation dla polynomially-ambiguous automatów ważonych.

(G) Ismaël Jecker, Filip Mazowiecki i David Purser.

Determinisation and Unambiguisation of Polynomially-Ambiguous Rational Weighted Automata.

Wersja konferencyjna *LICS* 2024 [50].

Zaczynamy od uwagi, że matematycznie bardziej naturalny wydaje się problem unambiguisation: czy dla danego automatu ważonego istnieje automat równoważny, ale jednoznaczny. Ponieważ determinisation jest znany jako rozstrzygalny dla automatów unambiguous, wystarczy skupić się na unambiguisation (choć w ogólności złożoność, może być inna co pomijamy tutaj). Ten wynik bazuje na tzw. twins property [29], którą łatwo można zaadaptować do automatów ważonych nad $\mathbb{Q}$.

Elegancja klasy unambiguous wynika stąd, że intuicyjnie są to automaty, które nie używają sumowania w nietrywialny sposób przy obliczaniu wyniku. Główne twierdzenie, na którym opieramy część wyników (G), pochodzi z [9]. Dla automatu $\mathcal{A}$ nad $\mathbb{Q}$ niech S_1 będzie zbiorem wartości wyjściowych, jak wcześniej. Autorzy dowodzą hipotezy Reutenauera, że $\mathcal{A}$ jest unambiguisable wtedy i tylko wtedy, gdy $S_1 \setminus \{0\}$ zawiera się w skończenie generowanej podgrupie (ze względu na operację mnożenia). Dla intuicji: jeśli $S_1 \subseteq \mathbb{Z}$, to oznacza to istnienie skończonego zbioru liczb pierwszych P , takiego że wszystkie dzielniki pierwsze niezerowych elementów S_1 należą do P . Jedna implikacja jest trywialna: mając automat unambiguous, bierzemy podgrupę generowaną przez niezerowe wagi na przejściach oraz w wektorach początkowych/końcowych. Trudność tkwi w dowodzie drugiej implikacji.

Nasza metoda opiera się na analizie zachowania $\mathcal{A}$ na rodzinach słów typu $(uw^nv)_{n \geq 1}$, dla $u, w, v \in \Sigma^*$. Ponieważ

$$\mathcal{A}(uw^nv) = I \cdot M(u) \cdot M(w)^n \cdot M(v) \cdot F,$$

zrozumienie tych zachowań sprowadza się do analizy potęg macierzy.

Prowadzi to do kluczowej definicji. Macierz M nazywamy p -triangular, jeśli istnieje macierz permutacyjna P taka, że $PM P^{-1}$ jest górnotrójkątna. Automat ważony $\mathcal{A}$ nazywamy *pumpable*, jeśli dla wszystkich $u, v, w \in \Sigma^*$, dla których $M(v)$ jest p -triangular, istnieje element diagonalny d macierzy $M(v)$ taki, że

$$\mathcal{A}(uv^{|A|+n}w) = d^n \cdot \mathcal{A}(uv^{|A|}w) \quad \text{dla każdego } n \in \mathbb{N}.$$

Zgodnie z intuicją z [9], automaty unambiguisable powinny być pumpable, ponieważ nie używają sumowania w nietrywialny sposób. Odwrotna implikacja nie jest prawdziwa w pełnej ogólności z dwóch powodów:

1. Pumpability odwołuje się do macierzy p -triangular, które mogą w ogóle nie wystąpić w danym automacie;
2. Pumpability gwarantuje periodyczność tylko na słowach zawierających wiele kopii pewnego fragmentu v ; nie jest jasne, jak przenieść tę strukturę na słowa pozbawione takich powtórzeń, np. słowa bezkwadratowe.

Ograniczamy się więc do automatów polynomially-ambiguous i pokazujemy, że jeśli taki automat jest pumpable, to jest również unambiguisable. W tym ustawieniu macierze o odpowiedniej strukturze idempotentnej są (po odpowiednim przestawieniu stanów) macierzami p -triangular, a ich występowanie zapewnia twierdzenie Ramseya.

Na koniec pokazujemy, że sprawdzenie pumpability można zredukować do testu równoważności automatów ważonych o wykładniczo większym rozmiarze. Daje to górne ograniczenie PSPACE. Co więcej, nasza procedura jest konstruktywna: jeśli wejściowy automat jest unambiguisable, konstruujemy równoważny automat unambiguous. Co zaskakujące, mimo relatywnie niskiej złożoności problemu decyzyjnego, automaty budowane przez nasz algorytm mogą mieć rozmiar ponad elementarny. Nie podajemy dolnych ograniczeń ani dla złożoności, ani dla rozmiaru automatu równoważnego.

5 Aktywność naukowa

Jestem współautorem 14 artykułów czasopismowych (2 opublikowanych przed obroną doktoratu) oraz 32 publikacji konferencyjnych (5 przed obroną). Jako główne osiągnięcia wybrałem 7 publikacji konferencyjnych [67, 68, 39, 22, 6, 37, 50], z których 5 posiada wersje czasopismowe [69, 25, 40, 23, 7]. Uważam je za swoje najciekawsze osiągnięcia związane z modelem automatów ważonych i jego wariantami.

Chronologicznie pracowałem nad tymi artykułami w następujących instytucjach:

- [67] i wersja czasopismowa [69]: praca powstała na Uniwersytecie Warszawskim po złożeniu rozprawy doktorskiej, publikacje ukazały się już po obronie;
- [68]: praca powstała podczas mojego postdoca na University of Warwick; wersja czasopismowa [25] została napisana we współpracy z dwoma nowymi autorami podczas mojego postdoca w Bordeaux;
- [39] i wersja czasopismowa [40]: prace powstały podczas mojego postdoca na Uniwersytecie Oksfordzkim;
- [22, 6] i wersje czasopismowe [23, 7]: prace powstały podczas mojego postdoca na Uniwersytecie w Bordeaux;
- [37]: praca powstała, gdy byłem na stanowisku research group leader w MPI-SWS;
- [50]: praca powstała po moim powrocie na Uniwersytet Warszawski.

Moja rozprawa doktorska dotyczyła logik związanych z teorią baz danych i opierała się na następujących artykułach czasopismowych [62, 12] oraz konferencyjnych [11, 61, 24, 63].

Po doktoracie pracowałem głównie nad dwoma modelami w weryfikacji formalnej: automatami ważonymi oraz sieciami Petriego. Wśród prac o automatach ważonych, których nie zaliczyłem do głównych osiągnięć, są artykuł czasopismowy [4] oraz materiały konferencyjne [66, 3, 30]. Mam też dwa nieopublikowane preprinty z tej dziedziny [41, 65].

Zdecydowana większość moich pozostałych prac dotyczy modelu sieci Petriego. Ze względu na odmienny charakter techniczny nie uwzględniłem ich w głównej części dokumentu. Wśród nich znajdują się artykuły czasopismowe [32, 36, 17, 19, 57] oraz publikacje konferencyjne [31, 43, 16, 34, 64, 35, 18, 20, 21, 56, 26, 70, 8, 49, 15, 27].

Jestem też współautorem dwóch prac które są nietypowe tematycznie względem pozostałych: jedna dotyczy struktur dynamicznych dla automatów czasowych [47] wraz z wersją czasopismową [48], druga — sieci neuronowych na grafach [46].

6 Dydaktyka, organizacja i popularyzacja nauki

Dydaktyka

- Uniwersytet Warszawski.

Koordynator:

- seminarium magisterskie: Języki automaty i obliczenia;
- cykl PhD open courses dla doktorantów i studentów magisterskich.

Ćwiczenia:

- Języki automaty i obliczenia;
- Logika dla informatyków;
- Wstęp do programowania;
- Automaty na nieskończonych słowach;
- Automaty ważone.

Wykłady:

- Logika dla informatyków;
- Automaty ważone.

- University of Warwick.

Laboratoria:

- Decision Procedures.

- Saarland University.

Wykłady:

- Weighted automata.

Organizacja

- współprzewodniczący komitetu organizacyjnego MFCS 2025 (local organiser co-chair);
- współorganizator LMW i LMW@CSL (Logic Mentoring Workshop at CSL) przy CSL'22, LICS'20, LICS'19;
- współorganizator warsztatu INFINITY przy LICS'20 i CONCUR'23;
- współorganizator warsztatu „Autobóz” o automatach, logice i grach, edycje z lipca 2018 i 2020;
- współorganizator seminarium zespołu „Formal Methods” w LaBRI od stycznia 2018 do lipca 2019.

Popularyzacja

- Był wywiad z nami w artykule oraz materiale wideo o sieciach Petriego w Quanta Magazine: <https://www.quantamagazine.org/an-easy-sounding-problem-yields-numbers-too-big-for-our-universe-20231204/>
https://www.youtube.com/watch?v=IzSs_gJDVzI
- Napisałem artykuł popularnonaukowy dla „DeltY”: <https://www.deltami.edu.pl/2018/11/jak-definiowac-ciagi-rekurencyjne/%0A>

7 Inne

- Byłem opiekunem jednego doktoranta: Philip Offtermatt (współpromotor: Michael Blondin). Obronił doktorat w 2023 roku na Université de Sherbrooke.
- Byłem opiekunem dwóch prac magisterskich: Aleksandra Wiśniewskiego (obrona w 2024) i Antoniego Pucha (obrona w 2025).

Literatura

- [1] Shaull Almagor, Guy Arbel, and Sarai Sheinvald. Determinization of min-plus weighted automata is decidable. *CoRR*, abs/2503.23826, 2025.
- [2] Shaull Almagor, Udi Boker, and Orna Kupferman. What’s decidable about weighted automata? *Information and Computation*, 282:1–20, 2022.
- [3] Shaull Almagor, Michaël Cadilhac, Filip Mazowiecki, and Guillermo A. Pérez. Weak cost register automata are still powerful. In Mizuho Hoshi and Shinnosuke Seki, editors, *Developments in Language Theory - 22nd International Conference, DLT 2018, Tokyo, Japan, September 10-14, 2018, Proceedings*, volume 11088 of *Lecture Notes in Computer Science*, pages 83–95. Springer, 2018.
- [4] Shaull Almagor, Michaël Cadilhac, Filip Mazowiecki, and Guillermo A. Pérez. Weak cost register automata are still powerful. *Int. J. Found. Comput. Sci.*, 31(6):689–709, 2020.
- [5] Rajeev Alur, Loris D’Antoni, Jyotirmoy V. Deshmukh, Mukund Raghothaman, and Yifei Yuan. Regular functions and cost register automata. In *28th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2013, New Orleans, LA, USA, June 25-28, 2013*, pages 13–22. IEEE Computer Society, 2013.
- [6] Corentin Barloy, Nathanaël Fijalkow, Nathan Lhote, and Filip Mazowiecki. A robust class of linear recurrence sequences. In Maribel Fernández and Anca Muscholl, editors, *28th EACSL Annual Conference on Computer Science Logic, CSL 2020, January 13-16, 2020, Barcelona, Spain*, volume 152 of *LIPIcs*, pages 9:1–9:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020.
- [7] Corentin Barloy, Nathanaël Fijalkow, Nathan Lhote, and Filip Mazowiecki. A robust class of linear recurrence sequences. *Inf. Comput.*, 289(Part):104964, 2022.

- [8] Pascal Baumann, Khushraj Madnani, Filip Mazowiecki, and Georg Zetsche. Monus semantics in vector addition systems with states. In Guillermo A. Pérez and Jean-François Raskin, editors, *34th International Conference on Concurrency Theory, CONCUR 2023, September 18-23, 2023, Antwerp, Belgium*, volume 279 of *LIPIcs*, pages 10:1–10:18. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023.
- [9] Jason P. Bell and Daniel Smertnig. Noncommutative rational pólya series. *Selecta Mathematica (N.S.)*, 27(3):Paper No. 34, 34, 2021.
- [10] Jason P. Bell and Daniel Smertnig. Computing the linear hull: Deciding DETERMINISTIC? and UNAMBIGUOUS? for weighted automata over fields. In *LICS '23: 38th Annual ACM/IEEE Symposium on Logic in Computer Science*, pages 1–13. ACM / IEEE, 2023.
- [11] Saguy Benaim, Michael Benedikt, Witold Charatonik, Emanuel Kieronski, Rastislav Lenhardt, Filip Mazowiecki, and James Worrell. Complexity of two-variable logic on finite trees. In Fedor V. Fomin, Rusins Freivalds, Marta Z. Kwiatkowska, and David Peleg, editors, *Automata, Languages, and Programming - 40th International Colloquium, ICALP 2013, Riga, Latvia, July 8-12, 2013, Proceedings, Part II*, volume 7966 of *Lecture Notes in Computer Science*, pages 74–88. Springer, 2013.
- [12] Saguy Benaim, Michael Benedikt, Witold Charatonik, Emanuel Kieronski, Rastislav Lenhardt, Filip Mazowiecki, and James Worrell. Complexity of two-variable logic on finite trees. *ACM Trans. Comput. Log.*, 17(4):32, 2016.
- [13] Yahia Idriss Benalioua, Nathan Lhote, and Pierre-Alain Reynier. Minimizing cost register automata over a field. In Rastislav Královic and Antonín Kucera, editors, *49th International Symposium on Mathematical Foundations of Computer Science, MFCS 2024, Bratislava, Slovakia, August 26-30, 2024*, volume 306 of *LIPIcs*, pages 23:1–23:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024.
- [14] Michael Benedikt, Timothy Duff, Aditya Sharad, and James Worrell. Polynomial automata: Zeroness and applications. In *32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2017, Reykjavik, Iceland, June 20-23, 2017*, pages 1–12. IEEE Computer Society, 2017.
- [15] Michael Blondin, Alain Finkel, Piotr Hofman, Filip Mazowiecki, and Philip Offtermatt. Soundness of reset workflow nets. In Pawel Sobocinski, Ugo Dal Lago, and Javier Esparza, editors, *Proceedings of the 39th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2024, Tallinn, Estonia, July 8-11, 2024*, pages 13:1–13:14. ACM, 2024.
- [16] Michael Blondin, Christoph Haase, and Filip Mazowiecki. Affine extensions of integer vector addition systems with states. In Sven Schewe and Lijun Zhang, editors, *29th International Conference on Concurrency Theory, CONCUR 2018, September 4-7, 2018, Beijing, China*, volume 118 of *LIPIcs*, pages 14:1–14:17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018.
- [17] Michael Blondin, Christoph Haase, Filip Mazowiecki, and Mikhail A. Raskin. Affine extensions of integer vector addition systems with states. *Log. Methods Comput. Sci.*, 17(3), 2021.
- [18] Michael Blondin, Tim Leys, Filip Mazowiecki, Philip Offtermatt, and Guillermo A. Pérez. Continuous one-counter automata. In *36th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2021, Rome, Italy, June 29 - July 2, 2021*, pages 1–13. IEEE, 2021.

- [19] Michael Blondin, Tim Leys, Filip Mazowiecki, Philip Offtermatt, and Guillermo A. Pérez. Continuous one-counter automata. *ACM Trans. Comput. Log.*, 24(1):3:1–3:31, 2023.
- [20] Michael Blondin, Filip Mazowiecki, and Philip Offtermatt. The complexity of soundness in workflow nets. In Christel Baier and Dana Fisman, editors, *LICS '22: 37th Annual ACM/IEEE Symposium on Logic in Computer Science, Haifa, Israel, August 2 - 5, 2022*, pages 20:1–20:13. ACM, 2022.
- [21] Michael Blondin, Filip Mazowiecki, and Philip Offtermatt. Verifying generalised and structural soundness of workflow nets via relaxations. In Sharon Shoham and Yakir Vizel, editors, *Computer Aided Verification - 34th International Conference, CAV 2022, Haifa, Israel, August 7-10, 2022, Proceedings, Part II*, volume 13372 of *Lecture Notes in Computer Science*, pages 468–489. Springer, 2022.
- [22] Michaël Cadilhac, Filip Mazowiecki, Charles Paperman, Michał Pilipczuk, and Géraud Sénizergues. On polynomial recursive sequences. In Artur Czumaj, Anuj Dawar, and Emanuela Merelli, editors, *47th International Colloquium on Automata, Languages, and Programming, ICALP 2020, July 8-11, 2020, Saarbrücken, Germany (Virtual Conference)*, volume 168 of *LIPIcs*, pages 117:1–117:17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020.
- [23] Michaël Cadilhac, Filip Mazowiecki, Charles Paperman, Michał Pilipczuk, and Géraud Sénizergues. On polynomial recursive sequences. *Theory of Computing Systems*, pages 1–22, 2021.
- [24] Witold Charatonik, Emanuel Kieronski, and Filip Mazowiecki. Decidability of weak logics with deterministic transitive closure. In Thomas A. Henzinger and Dale Miller, editors, *Joint Meeting of the Twenty-Third EACSL Annual Conference on Computer Science Logic (CSL) and the Twenty-Ninth Annual ACM/IEEE Symposium on Logic in Computer Science (LICS), CSL-LICS '14, Vienna, Austria, July 14 - 18, 2014*, pages 29:1–29:10. ACM, 2014.
- [25] Agnishom Chattopadhyay, Filip Mazowiecki, Anca Muscholl, and Cristian Riveros. Pumping lemmas for weighted automata. *Log. Methods Comput. Sci.*, 17(3), 2021.
- [26] Dmitry Chistikov, Wojciech Czerwinski, Piotr Hofman, Filip Mazowiecki, and Henry Sinclair-Banks. Acyclic petri and workflow nets with resets. In Patricia Bouyer and Srikanth Srinivasan, editors, *43rd IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2023, December 18-20, 2023, IIIT Hyderabad, Telangana, India*, volume 284 of *LIPIcs*, pages 16:1–16:18. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023.
- [27] Dmitry Chistikov, Wojciech Czerwinski, Filip Mazowiecki, Lukasz Orlikowski, Henry Sinclair-Banks, and Karol Węgrzycki. The tractability border of reachability in simple vector addition systems with states. In *65th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2024, Chicago, IL, USA, October 27-30, 2024*, pages 1332–1354. IEEE, 2024.
- [28] Dmitry Chistikov, Stefan Kiefer, Andrzej S. Murawski, and David Purser. The big-o problem. *Log. Methods Comput. Sci.*, 18(1), 2022.
- [29] Christian Choffrut. Une caractérisation des fonctions séquentielles et des fonctions sous-séquentielles en tant que relations rationnelles. *Theoretical Computer Science*, 5(3):325–337, 1977.

- [30] Lorenzo Clemente, Maria Donten-Bury, Filip Mazowiecki, and Michal Pilipczuk. On rational recursive sequences. In Petra Berenbrink, Patricia Bouyer, Anuj Dawar, and Mamadou Moustapha Kanté, editors, *40th International Symposium on Theoretical Aspects of Computer Science, STACS 2023, March 7-9, 2023, Hamburg, Germany*, volume 254 of *LIPIcs*, pages 24:1–24:21. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023.
- [31] Lorenzo Clemente, Slawomir Lasota, Ranko Lazic, and Filip Mazowiecki. Timed pushdown automata and branching vector addition systems. In *32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2017, Reykjavik, Iceland, June 20-23, 2017*, pages 1–12. IEEE Computer Society, 2017.
- [32] Lorenzo Clemente, Slawomir Lasota, Ranko Lazic, and Filip Mazowiecki. Binary reachability of timed-register pushdown automata and branching vector addition systems. *ACM Trans. Comput. Log.*, 20(3):14:1–14:31, 2019.
- [33] Anne Condon and Richard J. Lipton. On the complexity of space bounded interactive proofs (extended abstract). In *30th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 462–467, Research Triangle Park, North Carolina, USA, 1989. IEEE Computer Society.
- [34] Wojciech Czerwinski, Slawomir Lasota, Ranko Lazic, Jérôme Leroux, and Filip Mazowiecki. The reachability problem for petri nets is not elementary. In Moses Charikar and Edith Cohen, editors, *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019, Phoenix, AZ, USA, June 23-26, 2019*, pages 24–33. ACM, 2019.
- [35] Wojciech Czerwinski, Slawomir Lasota, Ranko Lazic, Jérôme Leroux, and Filip Mazowiecki. Reachability in fixed dimension vector addition systems with states. In Igor Konnov and Laura Kovács, editors, *31st International Conference on Concurrency Theory, CONCUR 2020, September 1-4, 2020, Vienna, Austria (Virtual Conference)*, volume 171 of *LIPIcs*, pages 48:1–48:21. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020.
- [36] Wojciech Czerwinski, Slawomir Lasota, Ranko Lazic, Jérôme Leroux, and Filip Mazowiecki. The reachability problem for petri nets is not elementary. *J. ACM*, 68(1):7:1–7:28, 2021.
- [37] Wojciech Czerwinski, Engel Lefauchaux, Filip Mazowiecki, David Purser, and Markus A. Whiteland. The boundedness and zero isolation problems for weighted automata over nonnegative rationals. In Christel Baier and Dana Fisman, editors, *LICS '22: 37th Annual ACM/IEEE Symposium on Logic in Computer Science, Haifa, Israel, August 2 - 5, 2022*, pages 15:1–15:13. ACM, 2022.
- [38] Laure Daviaud. Register complexity and determinisation of max-plus automata. *ACM SIGLOG News*, 7(2):4–14, 2020.
- [39] Laure Daviaud, Marcin Jurdzinski, Ranko Lazic, Filip Mazowiecki, Guillermo A. Pérez, and James Worrell. When is containment decidable for probabilistic automata? In Ioannis Chatzigiannakis, Christos Kaklamanis, Daniel Marx, and Donald Sannella, editors, *45th International Colloquium on Automata, Languages, and Programming, ICALP 2018, July 9-13, 2018, Prague, Czech Republic*, volume 107 of *LIPIcs*, pages 121:1–121:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018.
- [40] Laure Daviaud, Marcin Jurdzinski, Ranko Lazic, Filip Mazowiecki, Guillermo A. Pérez, and James Worrell. When are emptiness and containment decidable for probabilistic automata? *J. Comput. Syst. Sci.*, 119:78–96, 2021.

- [41] Karolina Drabik, Anita Dürr, Fabian Frei, Filip Mazowiecki, and Karol Wegrzycki. Fined-grained complexity of ambiguity problems on automata and directed graphs. *CoRR*, abs/2501.14725, 2025.
- [42] Manfred Droste and Paul Gastin. Weighted automata and weighted logics. *Theoretical Computer Science*, 380(1–2):69–86, 2007.
- [43] Diego Figueira, Ranko Lazic, Jérôme Leroux, Filip Mazowiecki, and Grégoire Sutre. Polynomial-space completeness of reachability for succinct branching VASS in dimension one. In Ioannis Chatzigiannakis, Piotr Indyk, Fabian Kuhn, and Anca Muscholl, editors, *44th International Colloquium on Automata, Languages, and Programming, ICALP 2017, July 10–14, 2017, Warsaw, Poland*, volume 80 of *LIPIcs*, pages 119:1–119:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017.
- [44] Nathanaël Fijalkow, Cristian Riveros, and James Worrell. Probabilistic automata of bounded ambiguity. In Roland Meyer and Uwe Nestmann, editors, *28th International Conference on Concurrency Theory, CONCUR 2017, September 5–8, 2017, Berlin, Germany*, volume 85 of *LIPIcs*, pages 19:1–19:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017.
- [45] Pierre Ganty and Elena Gutiérrez. The parikh property for weighted context-free grammars. In *38th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2018)*, volume 122 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 32:1–32:20, Dagstuhl, Germany, 2018. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.
- [46] Floris Geerts, Filip Mazowiecki, and Guillermo A. Pérez. Let’s agree to degree: Comparing graph convolutional networks in the message-passing framework. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18–24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 3640–3649. PMLR, 2021.
- [47] Alejandro Grez, Filip Mazowiecki, Michal Pilipczuk, Gabriele Puppis, and Cristian Riveros. Dynamic data structures for timed automata acceptance. In Petr A. Golovach and Meirav Zehavi, editors, *16th International Symposium on Parameterized and Exact Computation, IPEC 2021, September 8–10, 2021, Lisbon, Portugal*, volume 214 of *LIPIcs*, pages 20:1–20:18. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021.
- [48] Alejandro Grez, Filip Mazowiecki, Michal Pilipczuk, Gabriele Puppis, and Cristian Riveros. Dynamic data structures for timed automata acceptance. *Algorithmica*, 84(11):3223–3245, 2022.
- [49] Piotr Hofman, Filip Mazowiecki, and Philip Offtermatt. Fast termination and workflow nets. In Constantin Enea and Akash Lal, editors, *Computer Aided Verification - 35th International Conference, CAV 2023, Paris, France, July 17–22, 2023, Proceedings, Part I*, volume 13964 of *Lecture Notes in Computer Science*, pages 132–155. Springer, 2023.
- [50] Ismaël Jecker, Filip Mazowiecki, and David Purser. Determinisation and unambiguisation of polynomially-ambiguous rational weighted automata. In Pawel Sobocinski, Ugo Dal Lago, and Javier Esparza, editors, *Proceedings of the 39th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2024, Tallinn, Estonia, July 8–11, 2024*, pages 46:1–46:13. ACM, 2024.

- [51] Manuel Kauers and Peter Paule. *The Concrete Tetrahedron: Symbolic Sums, Recurrence Equations, Generating Functions, Asymptotic Estimates*. Texts & Monographs in Symbolic Computation. Springer Vienna, 2011.
- [52] Daniel Kirsten. A burnside approach to the termination of mohri’s algorithm for polynomially ambiguous min-plus-automata. *RAIRO Theor. Informatics Appl.*, 42(3):553–581, 2008.
- [53] Daniel Kirsten and Sylvain Lombardy. Deciding unambiguity and sequentiality of polynomially ambiguous min-plus automata. In Susanne Albers and Jean-Yves Marion, editors, *26th International Symposium on Theoretical Aspects of Computer Science, STACS 2009, February 26-28, 2009, Freiburg, Germany, Proceedings*, volume 3 of *LIPIcs*, pages 589–600. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, Germany, 2009.
- [54] Ines Klimann, Sylvain Lombardy, Jean Mairesse, and Christophe Prieur. Deciding unambiguity and sequentiality from a finitely ambiguous max-plus automaton. *Theor. Comput. Sci.*, 327(3):349–373, 2004.
- [55] Stephan Kreutzer and Cristian Riveros. Quantitative monadic second-order logic. In *28th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS 2013)*, pages 113–122. IEEE, 2013.
- [56] Marvin Künnemann, Filip Mazowiecki, Lia Schütze, Henry Sinclair-Banks, and Karol Wegrzycki. Coverability in VASS revisited: Improving rackoff’s bound to obtain conditional optimality. In Kousha Etessami, Uriel Feige, and Gabriele Puppis, editors, *50th International Colloquium on Automata, Languages, and Programming, ICALP 2023, July 10-14, 2023, Paderborn, Germany*, volume 261 of *LIPIcs*, pages 131:1–131:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023.
- [57] Marvin Künnemann, Filip Mazowiecki, Lia Schütze, Henry Sinclair-Banks, and Karol Wegrzycki. Coverability in VASS revisited: Improving rackoff’s bounds to obtain conditional optimality. *J. ACM*, 72(5):33:1–33:27, 2025.
- [58] Sylvain Lombardy. Sequentialization and unambiguity of $(\max, +)$ rational series over one letter. In *Workshop on max-plus algebras and Their Applications to Discrete-event Systems, Theoretical Computer Science, and Optimization*, page 6pp. IFAC, Elsevier Sciences, 2001.
- [59] Sylvain Lombardy and Jacques Sakarovitch. Sequential? *Theor. Comput. Sci.*, 356(1-2):224–244, 2006.
- [60] Angus Macintyre and Alex J. Wilkie. On the decidability of the real exponential field. In P. Odifreddi, editor, *Kreiseliana: About and Around Georg Kreisel*, pages 441–467. A K Peters, 1996.
- [61] Filip Mazowiecki, Filip Murlak, and Adam Witkowski. Monadic datalog and regular tree pattern queries. In Erzsébet Csuhaj-Varjú, Martin Dietzfelbinger, and Zoltán Ésik, editors, *Mathematical Foundations of Computer Science 2014 - 39th International Symposium, MFCS 2014, Budapest, Hungary, August 25-29, 2014. Proceedings, Part I*, volume 8634 of *Lecture Notes in Computer Science*, pages 426–437. Springer, 2014.
- [62] Filip Mazowiecki, Filip Murlak, and Adam Witkowski. Monadic datalog and regular tree pattern queries. *ACM Trans. Database Syst.*, 41(3):20:1–20:43, 2016.

- [63] Filip Mazowiecki, Joanna Ochremiak, and Adam Witkowski. Eliminating recursion from monadic datalog programs on trees. In Giuseppe F. Italiano, Giovanni Pighizzini, and Donald Sannella, editors, *Mathematical Foundations of Computer Science 2015 - 40th International Symposium, MFCS 2015, Milan, Italy, August 24-28, 2015, Proceedings, Part I*, volume 9234 of *Lecture Notes in Computer Science*, pages 394–406. Springer, 2015.
- [64] Filip Mazowiecki and Michal Pilipczuk. Reachability for bounded branching VASS. In Wan J. Fokkink and Rob van Glabbeek, editors, *30th International Conference on Concurrency Theory, CONCUR 2019, August 27-30, 2019, Amsterdam, the Netherlands*, volume 140 of *LIPICs*, pages 28:1–28:13. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019.
- [65] Filip Mazowiecki, Antoni Puch, and Daniel Smertnig. Pumping-like results for copyless cost register automata and polynomially ambiguous weighted automata. *CoRR*, abs/2502.07356, 2025.
- [66] Filip Mazowiecki and Cristian Riveros. Maximal partition logic: Towards a logical characterization of copyless cost register automata. In Stephan Kreutzer, editor, *24th EACSL Annual Conference on Computer Science Logic, CSL 2015, September 7-10, 2015, Berlin, Germany*, volume 41 of *LIPICs*, pages 144–159. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2015.
- [67] Filip Mazowiecki and Cristian Riveros. Copyless cost-register automata: Structure, expressiveness, and closure properties. In Nicolas Ollinger and Heribert Vollmer, editors, *33rd Symposium on Theoretical Aspects of Computer Science, STACS 2016, February 17-20, 2016, Orléans, France*, volume 47 of *LIPICs*, pages 53:1–53:13. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016.
- [68] Filip Mazowiecki and Cristian Riveros. Pumping lemmas for weighted automata. In Rolf Niedermeier and Brigitte Vallée, editors, *35th Symposium on Theoretical Aspects of Computer Science, STACS 2018, February 28 to March 3, 2018, Caen, France*, volume 96 of *LIPICs*, pages 50:1–50:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018.
- [69] Filip Mazowiecki and Cristian Riveros. Copyless cost-register automata: Structure, expressiveness, and closure properties. *J. Comput. Syst. Sci.*, 100:1–29, 2019.
- [70] Filip Mazowiecki, Henry Sinclair-Banks, and Karol Wegrzycki. Coverability in 2-vass with one unary counter is in NP. In Orna Kupferman and Pawel Sobocinski, editors, *Foundations of Software Science and Computation Structures - 26th International Conference, FoSSaCS 2023, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2023, Paris, France, April 22-27, 2023, Proceedings*, volume 13992 of *Lecture Notes in Computer Science*, pages 196–217. Springer, 2023.
- [71] Anca Muscholl and Gabriele Puppis. The many facets of string transducers (invited talk). In Rolf Niedermeier and Christophe Paul, editors, *36th International Symposium on Theoretical Aspects of Computer Science, STACS 2019, March 13-16, 2019, Berlin, Germany*, volume 126 of *LIPICs*, pages 2:1–2:21. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019.
- [72] Michael S Paterson. Unsolvability in 3×3 matrices. *Studies in Applied Mathematics*, 49(1):105–107, 1970.
- [73] Azaria Paz. *Introduction to probabilistic automata*. Academic Press, 1971.

- [74] Charles Rackoff. The covering and boundedness problems for vector addition systems. *Theoretical Computer Science*, 6:223–231, 1978.
- [75] Marcel Paul Schützenberger. On the definition of a family of automata. *Inf. Control.*, 4(2-3):245–270, 1961.
- [76] Géraud Sénizergues. Sequences of level 1, 2, 3, ..., k , .. In Volker Diekert, Mikhail V. Volkov, and Andrei Voronkov, editors, *Computer Science - Theory and Applications, Second International Symposium on Computer Science in Russia, CSR 2007, Ekaterinburg, Russia, September 3-7, 2007, Proceedings*, volume 4649 of *Lecture Notes in Computer Science*, pages 24–32. Springer, 2007.
- [77] Imre Simon. Factorization forests of finite height. *Theoretical Computer Science*, 72(1):65–94, 1990.
- [78] Andreas Weber and Helmut Seidl. On the degree of ambiguity of finite automata. *Theor. Comput. Sci.*, 88(2):325–349, 1991.