Rozprawa doktorska

Lokalne własności struktur losowych w teorii unikania wzorców

Adam Gągol
Uniwersytet Marii Curie-Skłodowskiej w Lublinie
6 grudnia 2017

Promotor: profesor Jarosław Grytczuk
Promotor pomocniczy: doktor Jakub Kozik

Streszczenie

Przedmiotem rozprawy doktorskiej są oszacowania na ilość kolorów w kolorowaniach unikających zadanego wzorca na grafach oraz słowach częściowych. W przypadku obu problemów wykorzystana jest metoda probabilistyczna oraz tak zwana kompresja entropii, czyli algorytmizacja Lokalnego Lematu Lovásza autorstwa Mosera i Tardosa.

Pierwszym badanym zagadnieniem jest szacowanie minimalnej długości wzorca o k zmiennych pozwalającej na unikanie go nad alfabetem binarnym oraz ternarnym. Rozważany jest wariant problemu, w którym słowo zawiera 'luki' - miejsca, które na potrzeby dopasowywania wzorca mogą zastępować dowolną literę. Przedstawiony wynik pokazuje oszacowanie równe oszacowaniu dla słów bez luk w przypadku wzorców o przynajmniej trzech zmiennych.

Drugim poruszanym zagadnieniem jest szacowanie minimalnej długości list, dla których możliwe jest pokolorowanie drzewa o zadanej szerokości ścieżkowej bez powtórzeń, tj. unikając wzorca AA. Wraz z oszacowaniem podany jest przykład klasy grafów o szerokości ścieżkowej 2 nie będących drzewami, dla których nie istnieje wspólne skończone oszacowanie na długość list pozwalającą na uniknięcie powtórzeń.

Podziękowania

W pierwszej kolejności chciałbym podziękować mojemu promotorowi, profesorowi Jarosławowi Grytczukowi, za zainteresowanie mnie problemami matematyki dyskretnej już na pierwszych latach studiów magisterskich i podsycanie tego zainteresowania przez niemal dziesięciolecie, czego rezultatem jest niniejsza rozprawa.

Chciałbym również podziękować dr. Jakubowi Kozikowi, bez pomocy którego nie powstałby prawdopodobnie żaden z prezentowanych tutaj wyników.

Podziękowania należą się również dr. Piotrowi Mickowi i dr. Gwenaëlowi Joretowi za wiele wspólnych sesji pracy naukowej, w wyniku których zrodził się wynik dotyczący unikania powtórzeń w kolorowaniach grafów.

Spis treści

1 Wstęp

Niniejsza rozprawa poświęcona jest zastosowaniu metody probabilistycznej w dowodzeniu twierdzeń kombinatorycznych dotyczących unikania wzorców.

Metoda probabilistyczna w kombinatoryce to szczególny sposób wykorzystania twierdzeń rachunku prawdopodobieństwa pozwalający dowodzić istnienie obiektów matematycznych spełniających określone właściwości. W swej podstawowej wersji zakłada konstrukcję odpowiedniej przestrzeni probabilistycznej i dowód, że losowo wybrany obiekt z tej przestrzeni z niezerowym prawdopodobieństwem posiada pożądane właściwości. Mimo że za pierwsze zastosowanie metody probabilistycznej uważa się zwykle pracę Szele dotyczącą cykli Hamiltona w turniejach z roku 1943 [24], jako głównego prekursora metody wymienia się zwykle Paula Erdősa, który prawdopodobnie jako pierwszy zdał sobie sprawę z jej ogólności i przez lata z powodzeniem stosował ją do różnorodnych problemów matematyki dyskretnej.

Ważnym wynikiem metody probabilistycznej jest Lokalny Lemat Lovásza [19] pozwalający na dowodzenie istnienia struktur unikających zbioru lokalnych własności. Istnieje wiele wersji lematu, historycznie pierwszą jest wersja symetryczna:

Lemat 1.1. *Niech $A_1, A_2, \ldots, A_k$ bądą zdarzeniami o prawdopodobieństwie nie większym od p takimi, że każde z nich jest całkowicie niezależne od wszystkich innych z wyjątkiem co najwyżej d pozostałych zdarzeń. Wtedy jeśli $ep(d+1) \leqslant 1$, to z niezerowym prawdopodobieństwem żadne ze zdarzeń $A_1, A_2, \ldots, A_k$ nie zachodzi.*

Własność lokalna w tym kontekście to taka, której możliwe realizacje w obrębie struktury są od siebie niezależne z wyłączeniem ograniczonej liczby przypadków, t.j. spełniają restrykcje nałożone na zdarzenia $A_1, A_2, \ldots, A_k$ w lemacie.

Pośród wielu wersji lematu istnieje również wersja algorytmiczna, w której za pomocą prostego algorytmu typu Las Vegas konstruowany jest obiekt unikający wszystkich nieporządanych własności lokalnych. Algorytm działa przy założeniu, że wszystkie zdarzenia A są determinowane przez skończoną liczbę niezależnych zmiennych losowych P. W ogólnej postaci algorytm przedstawia się następująco:

---

1 $\forall_{p \in P} : v_p \leftarrow$ losowa ewaluacja P
2 **while** $\exists t$ *takie, że* A_t *jest spełnione przez* $(v_p)_P$ **do**
3 wybierz dowolne zachodzące zdarzenie $A_{t'}$
4 $\forall_{p \in vbl(A_t) : v_p} \leftarrow$ nowa losowa ewaluacja
5 **return** $(v_p)_P$

W pierwszym kroku algorytm przypisuje losowe wartości wszystkim zmienym losowym $p \in P$. Następnie w głównej pętli losowo zmienia wartości zmiennych losowych składających się na zachodzące zdarzenia A aż do osiągnięcia sytuacji, w której żadne ze zdarzeń nie zachodzi. Poza oczywistym atutem takiego podejścia, czyli bezpośredniej konstrukcji obiektów spełniających dane zależności, pozwala ono rówież na poprawienie niektórych oszacowań wynikających z niealgorytmicznych wersji lematu. Jest to możliwe dzięki skorzystaniu z relacji między zmiennmi losowymi p a unikanymi zdarzeniami A, co wymyka się klasycznej wersji lematu.

Dwa przedstawione w rozprawie wyniki dotyczące unikania wzorców w słowach (rozdział 2) oraz grafach (rozdział 3) istotnie korzystają z tej relacji oraz algorytmicznej wersji lematu.

1.1 Oryginalne wyniki prezentowane w pracy

Główną treścią rozprawy będą dowody dwóch twierdzeń dotyczących unikania wzorców. Twierdzenia bedą wypowiedziane z pełnym formalizmem matematycznym w kolejnych rozdziałach, po wprowadzeniu odpowiednich definicji, tutaj zostaną jedynie wstępnie zarysowane. Pierwsze twierdzenie jest wynikiem samodzielnej pracy autora [8] i dotyczy unikania wzorców (tj. takich ciągów znaków, w których poszczególne podciągi mogą zostać przypisane kolejno po sobie występującym zmiennym we wzorcu) w słowach częściowych. Pojęcie $*$-unikalności występujące w wypowiedzi twierdzenia jest naturalnym rozszerzeniem unikalności na przypadek słów częściowych:

Twierdzenie 1.2. *Jeśli p jest wzorcem o $m > 2$ zmiennych takim, że $|p| \geqslant 2^m$, to p jest $*$-unikalny nad alfabetem ternarnym.*

Twierdzenie to przybliża pełną klasyfikację wzorców na takie, które są unikalne w słowach częściowych nad alfabetem binarnym, ternarnym i takie, które nie są unikalne nad żadnym skończonym alfabetem zostawiając problem otwartym jedynie dla trzech wzorców o dwóch zmiennych.

Drugim prezentowanym wynikiem będzie dowód twierdzenia:

Twierdzenie 1.3. *Istnieje funkcja f taka, że jeśli T jest drzewem o szerokości ścieżkowej w, to istnieje nierepetytywne kolorowanie T z list o długości $f(w)$.*

Razem z nim zaprezentowany będzie przykład klasy grafów o szerokości ścieżkowej 2, dla której nie istnieje skończone k takie, że każdy z grafów z tej klasy da się pokolorować z list długości k. Twierdzenie to zostało udowodnione we współpracy z Jakubem Kozikiem, Piotrem Mickiem oraz Gwenaëlem Joretem [9].

1.2 Wprowadzenie do metod analitycznych

Zanim przystąpimy do konstrukcji dowodów, przypomnimy klika podstawowych konceptów kombinatoryki analitycznej, dzięki którym uzyskamy później porządane oszacowania. Większość przedstawionych definicji i twierdzeń pochodzi z książki Philippe Flajoleta i Roberta Sedgewicka „Analytic combinatorics" [10].

Ciąg liczbowy $(a_i)_{i\in\mathbb{N}}$ jest *wykładniczego rzędu* K^n, co zapiszemy w skrócie jako $a_n \bowtie K^n$, wtedy i tylko wtedy gdy:

$$\limsup_{n\to\infty} \sqrt[n]{|a_n|} = K.$$

W dowodach użyjemy zdefiniowanych w książce operatorów działających na funkcjach tworzących SEQ oraz SEQ_k. Pierwszy z nich odpowiada klasie obiektów $1 + E + E^2 + \ldots$ i reprezentuje ciągi elementów z E, tzn. pozycje nie są permutowane i istnieje dokładnie jeden ciąg pusty. Mamy:

$$SEQ(f(z)) = 1 + \Sigma_{n\geqslant 1} f(z)^n = \frac{1}{1 - f(z)},$$

gdzie f jest funkcją tworzącą dla elementów z E. Operator SEQ_k odpowiada klasie obiektów E^k i reprezentuje ciągi o określonej długości, mamy zatem:

$$SEQ_k(f(z)) = f(z)^k$$

Kombinatoryka analityczna będzie używana w dowodach jako narzędzie służące do dowodzenia ograniczeń na wzrost współczynników funkcji tworzącej $f(z)$ zdefiniowanej równaniem typu $f(z) = z \cdot \phi(f(z))$. Będzie to możliwe dzięki zastosowaniu następującego twierdzenia:

Twierdzenie 1.4. *([10], Proposition IV.5) Niech ϕ będzie funkcją analityczną w 0, mającą nieujemne współczynniki Taylora taką, że $\phi(0) \neq 0$. Ponadto niech $R \leqslant +\infty$ będzie promieniem zbieżności ciągu reprezentującego ϕ*

w 0. *Wtedy przy zachowaniu warunku*

$$\lim_{x \to R_-} \frac{x\phi'(x)}{\phi(x)} > 1,$$

istnieje jednoznacznie wyznaczone rozwiązanie $\tau \in (0, R)$ *równania charakterystycznego:*

$$\frac{\tau\phi'(\tau)}{\phi(\tau)} = 1.$$

Ponadto formalne rozwiązanie $f(z)$ *równania* $f(z) = z \cdot \phi(f(z))$ *jest analityczne w* 0, *a jego współczynniki spełniają równanie wzrostu wykładniczego:*

$$[z^n]f(z) \bowtie (\frac{1}{p})^n \text{ gdzie } p = \frac{\tau}{\phi(\tau)} = \frac{1}{\phi'(\tau)}.$$

2 Unikanie wzorców w słowach częściowych

2.1 Podstawowe definicje

Niech $\Sigma = \{a, b, c, \dots\}$ oraz $\Delta = \{A, B, C, \dots\}$ będą alfabetami skończonymi. Elementy Σ nazywać będziemy *literami*, a ciągi elementów Σ *słowami*. Analogicznie elementy Δ nazywać będziemy *zmiennymi*, a ciągi tych elementów - *wzorcami*. Symbolem Σ^+ oznaczamy zbiór wszystkich słów skończonych (niepustych) nad alfabetem Σ. Podobnie Δ^+ oznacza zbiór wszystkich niepustych skończonych wzorców. Oba te zbiory możemy traktować jako monoidy z operacją sklejania słów.

Słowo w nad Σ *realizuje wzorzec* p jeśli istnieje niewymazujący morfizm $h : \Delta^+ \to \Sigma^+$ taki, że $h(p) = w$. Innymi słowy, jeśli istnieje podstawienie niepustych słów w miejsce zmiennych dające słowo w. Słowa realizujace wzorzec AA bedziemy nazywać *powtórzeniami*.

Słowo w *zawiera* wzorzec p wtedy i tylko wtedy, gdy jakieś jego spójne podsłowo realizuje p. Słowo w *unika* wzorca p jeśli go nie zawiera. Na przykład *aabaac* zawiera wzorzec ABA, podczas gdy *abaca* unika wzorca AA. Wzorzec p nazywamy *unikalnym* jeśli istnieje nieskończenie wiele słów nad pewnym alfabetem skończonym unikających p (lub równoważnie, jeśli istnieje nieskończone słowo unikające p).

Słowo częściowe nad alfabetem Σ to ciąg znaków z rozszerzonego alfabetu $\Sigma_\diamond = \Sigma \cup \{\diamond\}$, gdzie $\diamond$ rozumiemy jako *lukę* reprezentującą nieznany znak. Dla słowa częściowego w zbiór pozycji luk oznaczamy jako $H(w)$. Słowa, któte nie są częściowe, nazywać będziemy słowami *pełnymi*. Słowo częściowe *zawiera* wzorzec p jeśli istnieje podstawienie pojedynczych liter z Σ w miejsca $H(w)$ takie, że słowo wynikowe zawiera p. W przeciwnym razie mówimy, że w *unika* p. Wzorzec p nazywamy $*$-*unikalnym* jeśli istnieje nieskończone słowo częściowe w nad $\Sigma_\diamond$ unikające p oraz takie, że $|H(w)| = \infty$. Dla przykładu, wzorzec AA jest unikalny nad alfabetem ternarnym, natomiast nie jest $*$-

unikalny nad żadnym alfabetem skończonym (bo możemy zawsze wstawić literę poprzedzającą lukę w tę lukę).

Indeks unikalności $\lambda(p)$ dla wzorca p to rozmiar najmniejszego alfabetu Σ takiego, że istnieje słowo nieskończone nad Σ unikające p. Analogicznie *indeks częściowej unikalności* $\lambda^*(p)$ to rozmiar najmniejszego alfabetu Σ takiego, że istnieje nieskończone słowo częściowe w nad $\Sigma_\diamond$ unikające p oraz takie, że $|H(w)| = \infty$.

2.2 Wprowadzenie

Koncepcje i techniki związane z wzorcami w słowach znajdują zastosowania w wielu dziedzinach teoretycznej i stosowanej informatyki. Algorytmy związane z rozpoznawaniem wzorców znajdują szerokie zastosowanie np. w genetyce lub biologii obliczeniowej [5]. Z drugiej strony, ciągi bez powtórzeń są często wykorzystywane do znajdowania kontrprzykładów w teorii języków bezkontekstowych, teorii grup, problemach kombinatorycznych na zbiorach częściowo uporządkowanych oraz dynamice symbolicznej [6].

Koncepcja unikalności i nieunikalności wzorców została wprowadzona przez Bean'a, Ehrenfeucht'a i McNulty'ego [2] oraz niezależnie przez Zimina [27]. Udowodnili oni jedno z podstawowych twierdzeń w tej tematyce:

Twierdzenie 2.1. *Niech p będzie wzorcem nad m zmiennymi. Wtedy p jest unikalny nad alfabetem pewnej skończonej wielkości k wtedy i tylko wtedy, gdy w_m unika p, gdzie w_m jest rekursywnie zdefiniowane jako $w_1 = A_1$ oraz $w_l = w_{l-1}A_l w_{l-1}$.*

Interesującym i nietrywialnym wnioskiem z tego twierdzenia jest fakt, że nieunikalność danego wzorca p jest rozstrzygalna, tj. istnieje algorytm, który w ograniczonym czasie jest w stanie zdecydować, czy dany wzorzec jest unikalny. Wciąż otwartym poozstaje problem rozstrzygalności o unikalnośc wzorca p nad alfabetem określonej wielkości k. Alternatywną wersją

tego problemu jest znalezienie minimalnego k takiego, że p jest unikalny nad alfabetem wielkości k (tj. znalezienie indeksu unikalności $\lambda(p)$). W kontekście słów pełnych unikalność wzorców unarnych była badana przez Thue'go [25, 26]: A jest nieunikalny, $\lambda(AA) = 3$, natomiast $\lambda(AAA) = 2$. Przypadek $m = 2$ również został całkowicie sklasyfikowany [4, 18, 22, 23]. Wynik tej klasyfikacji można opisać następującym twierdzeniem [4]:

Twierdzenie 2.2. *Wzorce binarne pod względem unikalności dzielą się na następujące kategorie:*

1. *Wzorce nieunikalne nad żadnym alfabetem:* ϵ, A, AB, ABA,
2. *Wzorce unialne nad alfabetem ternarnym ale nieunikalne nad alfabetem binarnym:* $A^2, A^2B, A^2BA, A^2B^2, ABAB, AB^2A, A^2BA^2, A^2BAB$,
3. *Wzorce unikalne nad alfabetem binarnym: reszta.*

W przypadku słów częściowych krótkie wzorce są niemożliwe do uniknięcia ze względu na możliwość podstawienia luki pod dowolną zmienną, przez co zarówno A jak i AA są trywialnie nieunikalne, natomiast $\lambda^*(AAA) = 2$ [20], tak samo jak w przypadku słów pełnych. W przypadku $m = 2$ również widać odpowiedniość między słowami częściowymi a pełnymi - gdy zabroni się podstawiania pod zmienne pojedynczych liter, klasyfikacja wzorców dla słów pełnych pozostaje prawdziwa również w przypadku słów częściowych [3].

W ogólniejszej sytuacji, gdy wzorzec składa się z m zmiennych, hipoteza Cassaigne'a dowiedziona przez Blanchet-Sadri i Woodhouse'a daje ścisłe ograniczenie na minimalną długość wzorca unikalnego nad binarnym i ternarnym alfabetem dla słów pełnych:

Twierdzenie 2.3. *Jeśli p jest wzorcem o m zmiennych, to:*

- *Jeśli $|p| \geqslant 3(2^{m-1})$, to p jest unikalny nad alfabetem binarnym,*

- *Jeśli* $|p| \geqslant 2^m$*, to p jest unikalny nad alfabetem ternarnym.*

Autorzy w pracy stawiają hipotezę mówiącą, że powyższy wynik dla słów binarnych powinien uogólniać się na słowa częściowe. W przypadku słów ternarnych sytuacja jest nieco bardziej skomplikowana. Spostrzeżenie Kriegera i Ochema [17], że jeśli zmienna występuje we wzorcu tylko raz to można pod nią podstawić dowolnie długi ciąg liter i sprowadzić problem do wyszukiwania dwóch mniejszych wzorców pozwala sprowadzić problem unikalności dowolnego wzorca do problemu unikalności wzorca zdublowanego, czyli takiego, w którym każda zmienna występuje conajmniej dwukrotnie. W przypadku wzorców ternarnych możliwa jest sytuacja, w której pojedynczo występujące zmienne dzielą wzorzec na podwzorce w taki sposób, że żaden z nich nie spełnia już równania z warunku Cassaigne'a. Przykładem może być wzorzec $\alpha\alpha\beta\alpha\alpha\gamma\alpha\alpha$ - jest on nieunikalny, gdyż do tego by słowo go zawierało wystarczy, by para $a\diamond$ wystąpiła w słowie trzykrotnie dla pewnej litery a. Z tego względu w pewien sposób naturalne wydaje się rozważanie w przypadku alfabetu ternarnego jedynie wzorców zdublowanych. Głównym wynikiem przedstawianym w tym rozdziale będzie dowód hipotezy dla alfabetu ternarnego z pewnym ograniczeniem na m:

Twierdzenie 2.4. *Jeśli p jest wzorcem o* $m > 2$ *zmiennych takim, że* $|p| \geqslant 2^m$*, to p jest* $*$*-unikalny nad alfabetem ternarnym.*

Twierdzenie naturalnie uzupełnia przedstawioną wcześniej klasyfikację unikalnych wzorców w słowach częściowych pozostawiając jedynie niewielką lukę - wciąż nie jest wiadomo, czy istnieje zdublowany wzorzec długości 4 i o 2 zmiennych, który byłby nieunikalny dla słów częściowych nad alfabetem ternarnym. Takie wzorce są jedynie trzy - $ABAB$, $ABBA$ oraz $AABB$, jednak sprawdzenie ich unikalności wymyka się używanym w tej pracy metodom probabilistycznym, których siła tkwi w zachowaniach asymptotycznych.

2.3 Dowód twierdzenia 2.4

Dowód będzie wykorzystywać algorytmiczną wersję lokalnego lematu Lovásza autorstwa Mosera i Tardosa [21]. Koncepcja zastosowania tej wersji lematu w dowodach dotyczących wzorców w słowach pochodzi od Grytczuka, Kozika i Micka [12]. Polega na założeniu, że losowy algorytm budujący słowo unikając wzorca nigdy nie ułoży słowa pewnej długości M i użycia tego faktu do skompresowania ciągu bitów w stopniu większym niż jest to możliwe.

Dowód. Ustalmy wzorzec p o długości 2^k i $k \geqslant 3$ zmiennych. Udowodnimy, że możliwe jest skonstruowanie słowa $W = w_1 \dots w_n$ nad alfabetem $\Sigma = \{a, b, c\}$ unikającego p i takiego, że $H(W) = \{w_i : 100 \mid i\}$. Przeanalizujmy randomizowany algorytm 1, który stara się losowo ułożyć słowo W (dla uproszczenia analizy zakładamy, że umieszcza on litery również w miejscach luk) i wycofuje wszystkie instancje wzorca p traktując pozycje $H(W)$ jako luki.

Algorithm 1: Unikanie wzorca p

```
1  wejście: C : ℕ → Σ = {a, b, c}
2  i ← 1, j ← 1
3  while j < n do
4  |  w_j ← C(i)
5  |  i++
6  |  j++
7  |  if Istnieje instancja R wzorca p kończąca się na w_j then
8  |  |  Niech W_R będzie zbiorem pozycji R
9  |  |  for k ∈ W_R do
10 |  |  |  usuń literę na pozycji w_k
11 |  |  j ← pierwsza pozycja w W_R
12 return W
```

Załóżmy nie wprost, że nie istnieje $n-$literowe słowo unikające p, a zatem algorytm nigdy się nie zatrzyma. Ustalmy pewną wejściową sekwencję C i prześledźmy pierwsze M kroków działania algorytmu. Działając algorytmem na sekwencji C stworzymy strukturę dokładnie opisującą zachowanie algorytmu. Struktura ta będzie zależeć jedynie od sekwencji C oraz wzorca p. Ponadto z konstrukcji będzie wynikać, że dla dwóch róznych sekwencji C i C' otrzymamy różne struktury. Naszą strukturą będzie piątka $(P, L, S, H, F) \in (\mathbb{P}, \mathbb{L}, \mathbb{S}, \mathbb{H}, \mathbb{F})$, gdzie:

1. $P = (p_1, \ldots, p_M)$ jest ciągiem liczb, gdzie p_i jest różnicą w ilości liter, z których składa się konstruowane słowo po i-tym i $(i-1)$-ym kroku algorytmu (zatem $p_i < 0$ w przypadku wycofania instancji wzorca oraz $p_i = 1$ w przeciwnym wypadku),

2. $L = (L_1, \ldots, L_s)$ jest ciągiem zbiorów liczb, gdzie $L_i = \{l_{i,1}, \ldots, l_{i,k-1}\}$ (k jest ilością zmiennych we wzorcu p), a $l_{i,j}$ jest liczbą liter przyporządkowanych do j-tej zmiennej w i-tym wycofanym powtórzeniu wzorca p podczas działania algorytmu. Liczba liter przyporządkowanych do k - tej zmiennej nie musi być dodatkowo zapisywana, gdyż będzie możliwa do odtworzenia na podstawie pozostałych zmiennych i długości wycofanego fragmentu.

3. $S = (s_1, \ldots, s_r)$ jest ciągiem powstałym z liter przyporządkowanych do poszczególnych zmiennych α, β, γ kolejnych wycofywanych instancji p. Zapisywane są litery, które są przyporządkowane do danej zmiennej w miejscu nie zajętym przez lukę. W przypadku, gdy pewna pozycja danej zmiennej jest zajmowana przez lukę w każdej instancji tej zmiennej, w miejscu tej pozycji zostaje zapisana dowolna litera.

4. $H = \{h_1, \ldots, h_{|H|}\}$ jest ciągiem liter koniecznych do odtworzenia liter przyporządkowanych lukom podczas działania algorytmu. Po każdej

retrakcji do H dodawane jest tyle liter, ile luk znajdowało się w wycofywanym fragmencie. Jest pewna redundancja między H i S, ale nie jest ona asymptotycznie istotna.

5. $F = (f_1, \ldots, f_{n'})$ (gdzie $n' < n$) jest sekwencją liter w słowie W po M krokach algorytmu.

Bezstratność kompresji. Dowiedziemy, że możliwe jest zrekonstruowanie pierwszych M elementów sekwencji wejściowej C na podstawie piątki (P, L, S, H, F) powstałej w trakcie M kroków algorytmu. Dowód będzie polegał na odtworzeniu z (P, L, S, H, F) elementu $C(M)$ oraz (P', L', S', H', F') - piątki powstałej w trakcie pierwszych $M - 1$ kroków działania algorytmu. Iterując ten proces otrzymamy całą sekwencję C.

Jeśli $p_M = 1$, to żaden wzorzec nie został wycofany podczas ostatniego kroku algorytmu, więc $C(M)$ jest po prostu ostatnim elementem F, P' oraz F' są o jeden element krótsze, natomiast $L' = L$ i $H' = H$.

Jeśli $p_M = 1 - r$, to w trakcie ostatniego kroku nastąpiło wycofanie r elementów będących instancją wzorca p. Z ostatniego elementu L jesteśmy w stanie odtworzyć stukturę tej instancji, tzn. długości $l_1, \ldots, l_k$ ciągów podstawionych pod kolejne zmienne. Następnie analizując P jesteśmy w stanie stwierdzić ile luk zawierała instancja oraz w których miejscach wycofanego fragmentu się one znajdowały, natomiast z S i H jesteśmy w stanie odtworzyć konkretne litery, które były podstawione pod zmienne. Połączenie tych kroków pozwala na zrekonstuowanie słowa sprzed retrakcji wykonanej przez algorytm w ostatnim kroku. Przypisanie zrekonstruowanego słowa do F' oraz odpowiednie skrócenie P', L', S' oraz H' daje nam porządaną piątkę odpowiadającą $(M - 1)$-emu krokowi algorytmu.

Kompresja. Interesować nas będzie asymptotyczna ilość piątek P, L, S, H, F, gdy M dąży do nieskończoności. Podamy wspólne ograniczenie na ilość trójek P, L, S i osobne ograniczenia na H i F.

Ograniczenie liczby trójek (P, L, S)**.** Zastosujemy opisane wcześniej metody analityczne do ograniczenia rzędu wykładniczego ciągu $(T_n)_{n\in\mathbb{N}}$ trójek (P, L, S), które możliwe są do uzyskania po n krokach algorytmu. Na potrzeby szacowania dokonamy dwóch niewpływających na asymptotykę modyfikacji trójek P, L, S:

1. Dla każdej trójki przedłużmy P w taki sposób, by kończyła się słowem pustym, tj. by $p_n = 0$. Ze względu na ograniczoną ilość możliwych wartości p_i da się to zrobić dopisując jedynie stałą liczbę elementów,

2. Na początku ciągu P dodamy dodatkowy wyraz o wartości 1.

W ten sposób uzyskamy ciągi P, których sumy pozątkowe nigdy nie osiągają wartości 0 oraz których całkowita suma wynosi 1. Zbiór takich ciągów nazwijmy $\mathbb{P}_1$. Zauważmy, że konsekwencją tych własności jest to, że $P \in \mathbb{P}_1$ jest albo długości 1, albo w ostatnim kroku następuje spadek (czyli $p_n < 0$). Podążając za ideą dekompozycji ostatnich spadków [10, rozdział V.4], udowodnimy teraz:

Fakt 2.5. *Funkcja tworząca zliczającego obiekty w* $\mathbb{P}_1$ *spełnia równanie*

$$\mathcal{P}_1(z) = z\Big(1 + \frac{1}{1 - \mathcal{P}_1}\Big).$$

Dowód. Niech P będzie ciągiem z $\mathbb{P}_1$ o długości n i ostatnim spadku wysokości d. Ponadto niech i_h będzie ostatnim indeksem, dla którego $\Sigma_{i=1}^{i_h} p_i = h$ oraz niech $i_0 = 0$. Wtedy P_h zdefiniujemy jako $[p_{i_{h-1}+1}, \ldots, p_{i_h}]$. Zauważmy teraz, że wszystkie ciągi P_h należą również do $\mathbb{P}_1$ i po konkatenacji tworzą pierwsze $n-1$ wyrazów ciągu P, zatem po skonkatenowaniu z jednowyrazowym ciągiem $[1-d]$ tworzą całe $\mathbb{P}$. Ze względu na jednoznaczność rozkładu na P_h oraz jednowyrazowy ciąg daje nam to równanie rekurencyjne dla funkcji tworzącej:

$$\mathcal{P}_1(z) = z\Big(\mathrm{SEQ}(\mathcal{P}_1(z))\Big) = z\Big(\frac{1}{1-\mathcal{P}_1(z)}\Big) = z\Big(1+\frac{\mathcal{P}_1(z)}{1-\mathcal{P}_1(z)}\Big). \quad (1)$$

□

Niech $\mathbb{PL}_1$ oznacza zbiór par (P, L), gdzie $P \in \mathbb{P}_1$ oraz $L \in \mathbb{L}$. Teraz by uzyskać funkcję tworzącą zliczającą takie pary należy zdefiniować funkcję, która poza opisem ciągów P wraz z każdym spadkiem będzie zapisywać również odpowiedni jego podział na k części (gdzie k to ilość różnych zmiennych we wzorcu):

Fakt 2.6. *Funkcja tworząca zliczającego obiekty w* $\mathbb{PL}_1$ *spełnia równanie*

$$\mathcal{PL}_1(z) = z\Big(1+\prod_{i=1}^{k}\frac{\mathcal{PL}_1(z)^{u_i}}{1-\mathcal{PL}_1(z)^{u_i}}\Big),$$

gdzie u_i *jest ilością wystąpień* i*-tej zmiennej we wzorcu.*

Dowód. Niech P, L będzie parą z $\mathbb{PL}_1$, gdzie P jest długości n i ma ostatni spadek wysokości d. Dekompozycja opisująca $\mathbb{PL}_1$ będzie podobnej postaci jak (1), jednak by dodatkowo opisać podział ostatniego spadku, należy podzielić składowe P_h na k kolejnych podzbiorów, z których każdy jest wielkości sumarycznej ilości liter przyporządkowanych do wszystkich wystąpień odpowiadającej mu zmiennej. Należy zwrócić uwagę na fakt, że ta wartość będzie podzielna przez ilość wystąpień zmiennej odpowiadającej danemu podzbiorowi w całym wzorcu. Wszystkie spadki w P nie licząc ostatniego są opisane poprzez inne elementy (P, L), zatem ich podział nie musi być dodatkowo opisywany w dekompozycji. Konstrukcję taką można uzyskać za pomocą operatora SEQ_k:

$$\mathcal{PL}_1(z) = z\Bigg(1+\prod_{i=1}^{k}\mathrm{SEQ}\Big(\mathrm{SEQ}_{u_i}(\mathcal{PL}_1(z))\Big)\Bigg) = z\Bigg(1+\prod_{i=1}^{k}\frac{\mathcal{PL}_1(z)^{u_i}}{1-\mathcal{PL}_1(z)^{u_i}}\Bigg). \quad (2)$$

□

Niech teraz $\mathbb{T}$ będzie zbiorem trójek P, L, S takich, że $P \in \mathbb{P}_1$, $L \in \mathbb{L}$ oraz $S \in \mathbb{S}$. By uzyskać funkcję tworzącą zliczającą elementy $\mathbb{T}$ należy powyższe konstrukcje uzupełnić tak, by zapisywały dodatkowo wszystkie zapisywane litery:

Fakt 2.7. *Funkcja tworząca zliczającego obiekty w $\mathbb{T}$ spełnia równanie*

$$\mathcal{T}(z) = z\Big(1 + \prod_{i=1}^{k} \frac{3 \cdot \mathcal{T}(z)^{u_i}}{1 - 3 \cdot \mathcal{T}(z)^{u_i}}\Big),$$

gdzie u_i jest ilością wystąpień i-tej zmiennej we wzorcu.

Dowód. Niech P, L, S będzie trójką z $\mathbb{T}$, gdzie P jest długości n i ma ostatni spadek wysokości d. Dekompozycja opisująca $\mathbb{T}$ będzie podobnej postaci jak 2, jednak dodatkowo będzie zapisywać wycofane litery. Można to osiągnąć dopisując po prostu wartość 3 przed symbolem SEQ_u, co jest równoważne skojarzeniu wartości 3 z każdym wystąpieniem $T(z)^{u_i}$ po prawej stronie równania. Konstrukcja taka odpowiada zapisywaniu tylu liter, ile w ostatnim spadku zostało przyporządkowanych do wszystkich zmiennych (zliczając tylko pojedyncze wystąpienia każdej z nich):

$$\mathcal{T}(z) = z\bigg(1+\prod_{i=1}^{k}\Big(\mathrm{SEQ}\big(3\cdot\mathrm{SEQ}_{u_i}(\mathcal{T}(z))\big)\Big)\bigg) = z\bigg(1+\prod_{i=1}^{k}\frac{3\cdot\mathcal{T}(z)^{u_i}}{1-3\cdot\mathcal{T}(z)^{u_i}}\bigg). \quad (3)$$

□

Funkcja $\phi(t) = 1 + \Big(\prod_{i=1}^{k}\big(\frac{3t^{u_i}}{1-3t^{u_i}}\big)\Big)$ spełnia warunki twierdzenia 1.4, a zatem istnieje jednoznaczne rozwiązanie $\tau \in (0, R)$ równania $\frac{\tau\phi'(\tau)}{\phi(\tau)} = 1$, gdzie R jest promieniem zbieżności ϕ. Zdefiniujmy $\psi(x) = \frac{x}{\phi(x)}$ i zauważmy, że

$$\psi(x)' = \frac{\phi(x) - x \cdot \phi'(x)}{\phi^2(x)} = \frac{1}{\phi(x)} - \frac{x\phi'(x)}{\phi^2(x)}.$$

Zatem mamy następujące równoważności między równaniami:

$$\psi'(x) = 0 \tag{4}$$

$$\Updownarrow$$

$$\frac{1}{\phi(x)} = \frac{x\phi'(x)}{\phi^2(x)}$$

$$\Updownarrow$$

$$\frac{x\phi'(x)}{\phi(x)} = 1. \tag{5}$$

Zauważając ponadto, że $\psi(0) = 0$ oraz że ψ przyjmuje wartości nieujemne w obrębie promienia zbieżności dochodzimy do wniosku, że rozwiązanie równania (5) jest maksimum funkcji ψ w tym obszarze. Jako że na mocy twierdzenia 1.4 interesująca nas asymptotyka $[z^n]\mathcal{T}(z)$ jest wykładniczego rzędu $\left(\frac{1}{p}\right)$, gdzie $p = \frac{\tau}{\phi(\tau)} = \psi(\tau)$, by otrzymać górne oszacowanie na wykładniczy rząd asymptotyczny współczynników rozwinięcia $\mathcal{T}$, wystarczy nam teraz oszacować od dołu maksimum $\psi(t)$. Jako że interesuje nas tylko oszacowanie, zawęzimy się w naszych rozważaniach do $0 < t < 0.5$, co pozwoli uniknąć problemów z zależnością mnięzy promieniem zbieżności $\mathcal{T}$ (a zatem i dziedziną ψ), a parametrami k i l_i, gdyż promień zbieżności w interesującym nas podzbiorze możliwych parametrów jest zawsze większy od 0.5, więc rozważane funkcje będą ściśle dodatnie i określone w całej dziedzinie. Znajdziemy najpierw takie parametry k oraz l_i, dla których maksimum ψ jest najmniejsze. Najpierw znajdziemy odpowiednie l_i przy ustalonym k:

Fakt 2.8. *Niech* $0 < t < 0.5$. *Wtedy funkcja*

$$\psi_t(l_1, \ldots, l_k) = \frac{t}{1 + \left(\prod_{i=1}^{k}\left(\frac{3t^{l_i}}{1-3t^{l_i}}\right)\right)}$$

o dziedzinie równej $L = \left\{l_1, \ldots, l_k : l_i \geqslant 2, \Sigma_{i=1}^{k} l_i = 2^k\right\}$ *przyjmuje minimum w jednym z punktów ekstremalnych dziedziny, tj. gdy wszystkie* l_i *są równe* 2 *z wyjątkiem co najwyżej jednego.*

Dowód. Niech $l_m = \frac{l_a+l_b}{2}$ oraz niech $t > 0$. Z wypukłości funkcji wykładniczej wynika wtedy następujący ciąg równoważności:

$$t^{l_a} + t^{l_b} \geqslant 2t^{l_m}$$

$$\Updownarrow$$

$$1 - 3t^{l_a} - 3t^{l_b} + 9t^{l_a+l_b} \leqslant 1 - 6t^{l_m} + 9t^{2l_m}$$

$$\Updownarrow$$

$$1 + C\frac{9t^{l_a+l_b}}{(1-3t^{l_a})(1-3t^{l_b})} \geqslant 1 + C\frac{9t^{2l_m}}{(1-3t^{l_m})(1-3t^{l_m})}$$

$$\Updownarrow$$

$$\frac{t}{1 + C\frac{9t^{l_a+l_b}}{(1-3t^{l_a})(1-3t^{l_b})}} \leqslant \frac{t}{1 + C\frac{9t^{l_i+l_j}}{(1-3t^{l_m})(1-3t^{l_m})}}$$

$$\Updownarrow$$

$$\frac{t}{1 + C\frac{3t^{l_a}}{(1-3t^{l_a})} \cdot \frac{3t^{l_b}}{(1-3t^{l_b})}} \leqslant \frac{t}{1 + C\frac{3t^{l_m}}{(1-3t^{l_m})} \cdot \frac{3t^{l_m}}{(1-3t^{l_m})}}$$

dla dowolnej stałej C. Ustalając C równe iloczynowi $\frac{3t^{l_i}}{(1-3t^{l_i})}$ dla wszystkich pozostałych l_i dostajemy wklęsłość funkcji ψ_t. Funkcja wklęsła na zbiorze wypukłym, jakim jest ustalona dziedzina l_i, przyjmuje minimum w jednym z punktów ekstremalnych tego zbioru, co daje tezę. □

Znając teraz zależność między wartościami l_i oraz k jesteśmy w stanie wybrać odpowiednie k:

Fakt 2.9. *Niech* $0 < t < 0.5$. *Wtedy*

$$\psi_t^{ext}(k) = \frac{t}{1 + \left(\left(\frac{3t^2}{1-3t^2}\right)^{k-1}\left(\frac{3t^{2^k-2k+2}}{1-3t^{2^k-2k+2}}\right)\right)}$$

jest funkcją ściśle rosnąca dla $0 < t < 0.5$ *oraz* k *naturalnego większego od* 2.

Dowód. Fakt ten można udowodnić licząc pochodną funkcji $\psi_t^{ext}(k)$ i sprawdzając jej dodatniość, jednak postać pochodnej jest tak skomplikowana, że unikamy tego podejścia kierując się klarownością wywodu. Przeanalizujmy wpływ zwiększenia k o jeden na wartość funkcji. Rozważymy go poprzez analizę trzech termów, w których występuje k: $\left(\frac{3t^2}{1-3t^2}\right)^{k-1}$(1), $3t^{2^k-2k+2}$(2) oraz $\frac{1}{1-3t^{2^k-2k+2}}$(3):

(1) $\frac{3t^2}{1-3t^2} < 3$, zatem pierwszy term zwiększy się najwyżej trzykrotnie,

(2) $\frac{3t^{2^{k+1}-2(k+1)+2}}{3t^{2^k-2k+2}} = t^{2^k-2}$, zatem drugi term zmaleje przynajmniej 32-krotnie,

(3) $1 < \frac{1}{1-3t^{2^k-2k+2}} < \frac{16}{15}$ dla każdego k, zatem trzeci term wzrośnie najwyżej $\frac{16}{15}$-krotnie.

Z powyższej analizy jasno wynika, ze mianownik $\psi_t^{ext}(k)$ jest funkcją malejącą, a zatem $\psi_t^{ext}(k)$ jest rosnąca.

□

Z faktów 2.8 i 2.9 wynika, że najmniejsze maksimum funkcja ψ przyjmuje dla parametrów $k = 3$ oraz $l_1 = 2, l_2 = 2, l_3 = 4$. Używając programu Maple dokonujemy sprawdzenia (stosując obliczenia symboliczne), że dla takich parametrów osiąga ona wartość większą niż 0.487 dla $z = 0.471$, a zatem rząd wykładniczy T_n jest nie większy niż $1/0.4710 = 2.1229$ i w konsekwencji istnieje najwyżej 2.1229^M trójek P, L, S możliwych do uzyskania w M krokach algorytmu 1.

Ograniczenie liczby elementów H. Każda retrakcja R wycofuje co najmniej 8 liter, a zatem dodaje co najwyżej $\lceil \frac{|R|}{100} \rceil$ liter do H. Ponadto suma długości wszystkich retrakcji jest niewiększa niż M, a zatem istnieje najwyżej $2^{\frac{M}{8}} < 1.1^M$ ciągów R możliwych do osiągnięcia po M krokach algorytmu.

Ograniczanie liczby elementów F. Ostateczna sekwencja liter jest długości najwyżej $|W| - 1$, zatem istnieje najwyżej $4^{|W|}$ takich sekwencji, bo do

każdego miejsca może zostać przyporządkowany jeden z symboli $0, 1, 2$ (litery alfabetu ternarnego) lub brak symbolu. Symbole $\diamond$ mogą być przyporządkowywane tylko do ustalonych wcześniej pozycji, zatem dodatkowe zapisywanie ich nie jest konieczne.

Ograniczanie liczby piątek (P, L, S, R, F)**.** Mnożąc wszystkie poprzednie ograniczenia dla dostatecznie dużego M otrzymujemy ograniczenie $(2.1229 \times 1.1)^M \times 4^{|W|} < 2.34^M \times 4^{|W|} < 3^M$ na wszystkie możliwe piątki w pełni opisujące wszystkie 3^M możliwych prefiksów C, co daje nam pożądaną sprzeczność. □

2.4 Uwagi końcowe

Algorytmiczny Lokalny Lemat Lovásza jest bardzo użyteczną metodą w dziedzinie unikania wzorców w słowach, jednak jego siła przejawia się głównie w zachowaniach asymptotycznych (można to zresztą powiedzieć o większości metod o naturze probabilitycznej). Bezpośrednie zastosowanie technik wykorzystanych w dowodzie dla alfabetów ternarnych pozwala na udowodnienie hipotezy o alfabetach binarnych dla wzorców p o conajmniej 3 zmiennych. Z tego względu pełny dowód hipotezy dla alfabetów binarnych może wymagać innych metod.

3 Unikanie wzorców w grafach

3.1 Podstawowe definicje

Dla grafu G przez $V(G)$ i $E(G)$ oznaczać będziemy odpowiednio zbiór wierzchołków i krawędzi G. *Dekompozycja drzewowa* grafu G to graf T będący drzewem (grafem spójnym nieposiadający cyklu) o wierzchołkach $X_1, \ldots, X_n$,

gdzie każde X_i jest podzbiorem wierzchołków grafu G spełniającym następujące warunki:

1. Każdy wierzchołek grafu G należy do przynajmniej jednego X_i.
2. Jeśli $v \in V(G)$ należy zarówno do X_i jak i X_j, to wszystkie wierzchołki $X_k \in V(T)$ leżące na (jedynej) ścieżce między X_i a X_j również zawierają v.
3. Dla każdej krawędzi $(v, w) \in E(G)$ istnieje X_i zawierający zarówno v jak i w.

Analogicznie *dekompozycją ścieżkową* będziemy nazywać graf P będący ścieżką o wierzchołkach $X_1, \ldots, X_n$ spełniających powyższe warunki. Dla ustalonej dekompozycji (drzewowej lub ścieżkowej) przez jej *szerokość* rozumieć będziemy rozmiar największego zbioru X_i. Minimalną możliwą szerokość dekompozycji drzewowej grafu G nazywać będziemy *szerokością drzewową* G, natomiast minimalną szerokość dekompozycji ścieżkowej G - jego *szerokością ścieżkową*.

Kolorowanie grafu to funkcja $\phi : V(G) \to \mathbb{N}$ przyporządkowująca każdemu z wierzchołków kolor będący liczbą naturalną, natomiast k-kolorowanie to kolorowanie używające jedynie k kolorów. Powiemy, że kolorowanie $\phi : V(G) \to \mathbb{N}$ grafu G *unika* wzorca p, jeśli nie istnieje w nim ścieżka prosta, która zawierałaby p (rozważamy kolory wierzchołków na ścieżce jako litery alfabetu k-arnego). W szczególności, kolorowanie jest *nierepetytywne* jeśli unika wzorca AA. *Liczba Thuego* grafu G, oznaczana jako $\pi(G)$, to minimalna liczba kolorów k, dla której da się skonstruować nierepetytywne k-kolorowanie G.

Przyporządkujmy teraz każdemu z wierzchołków v listę $L_v \subset \mathbb{N}$ dostępnych kolorów. Kolorowanie G z list to takie kolorowanie, dla którego $\forall_{v \in V(G)} \phi(v) \in L_v$. *Listowa liczba Thuego* grafu G, oznaczana jako $\pi_l(G)$,

to minimalne k takie, że dla każdego przyporządkowania list wielkości k do wierzchołków G istnieje nierepetytywne kolorowanie z tych list.

W obrębie tego rozdziału interesować nas będzie czasem wybieranie mniejszej listy z większych list kolorów dostępnych dla każdego wierzchołka - na taką procedurę patrzeć będziemy jak na wybór podlisty ze wszystkich możliwych podlist i nazywać *prekolorowaniem*. O prekolorowaniu powiemy, że unika wzorca p, jeśli z wybranych list nie da się wybrać kolorów tak, by powstała instancja wzorca.

3.2 Wprowadzenie

Jednym z naturalnych rozwinięć tematyki unikania wzorców w słowach nad alfabetem k-arnym jest unikanie wzorców w grafach kolorowanych k kolorami. Problem ten był rozważany głównie w kontekście unikania powtórzeń (tzn. wzorców typu AA) i ograniczania liczby kolorów ze względu na różne parametry grafowe. Wynikiem, który zapoczątkował badania w tej dziedzinie było ograniczenie ilości kolorów ze względu na maksymalny stopień wierzchołka [1]:

Twierdzenie 3.1. *Istnieje stała c taka, że*

$$\pi(G) \leqslant \pi_l(G) \leqslant c\Delta^2$$

dla wszystkich grafów o maksymalnym stopniu wierzchołka równym Δ.

Ponadto w tej samej pracy udało się pokazać, że istnieją grafy, dla których $\pi(G) \geqslant \frac{\Delta^2}{\log \Delta}$. Dzięki późniejszym pracom dotyczącym oszacowania stałej c [13, 14, 15, 7] zostało osiągnięte oszacowanie $\pi(G) \leqslant \pi_l(G) \leqslant (1 + o(1))\Delta^2$ [7]. Wszystkie te wyniki w naturalny sposób uogólniają się na listową liczbę Thuego.

Jednym z prostszych wyników w tej dziedzinie jest ograniczenie liczby Thuego dla wszystkich drzew przez 4 [1], które zostało uogólnione dla wszyst-

kich grafów o ograniczonej szerokości drzewowej przez Kündgena i Pelsmajera. Pokazali oni, że $\pi(G) \leqslant 4^k$ dla wszystkich grafów o szerokości drzewowej k [16]. Jeśli zamiast szerokości drzewowej rozważy się szerokość ścieżkową, znane jest ograniczenie wielomianowe: $\pi(G) \leqslant 2k^2 + 6k + 1$ [7].

Przejdziemy teraz do omówienia wyników dotyczących listowej liczby Thuego, która będzie głównym tematem tego rozdziału. Wiadomo, że dla ścieżek listowa liczba Thuego wynosi 3 lub 4 i jest to jeden z pierwszych wyników wykorzystujących algorytmiczną wersję Lokalnego Lematu Lovásza w problematyce unikania wzorców [12] (zwykła liczba Thuego dla ścieżek wynosi 3 i jest to prosta konsekwencja twierdzenia Thuego). Niedawny wynik Fiorenzi'ego, Ochema, Ossona de Mendeza i Zhu jest pierwszym wskazującym na różnicę między listową i klasyczna liczbą Thuego [11]. Pokazali oni, że listowa liczba Thuego dla drzew nie jest ograniczona. Oczywiście drzewa, które osiągają wysoką listową liczbę Thuego muszą mieć również wysokie stopnie wierzchołków, co wynika z przytoczonego wcześniej ograniczenia.

Z jednej strony wiemy więc, że $\pi_l(G)$ nie jest ograniczone dla grafów z ograniczoną szerokością drzewową, z drugiej strony jest ograniczone dla ścieżek i grafów z szerokością ścieżkową 1 [7]. Dwa wyniki, które zaprezentuję w tym rozdziale powstały we współpracy z Piotrem Mickiem, Jakubem Kozikiem oraz Gwenaëlem Joretem. Pierwszy z nich to konstrukcja kontrprzykładu pokazującego, że listowa liczba Thuego nie jest ograniczona dla grafów z organiczoną szerokością ścieżkową:

Twierdzenie 3.2. *Dla każdego $c > 1$ istnieje graf G o szerokości ścieżkowej 2 taki, że $\pi_l(G) > c$.*

W obliczu tego wyniku i prostego ograniczenia klasycznej liczby Thuego dla drzew zasadnym wydaje się pytanie o listową liczbę Thuego dla drzew z ograniczoną szerokością ścieżkową (ograniczenie to nie ogranicza stopni wierzchołków). Drugi wynik jest pozytywny i stanowi ograniczenie na liczbę kolorów w tej właśnie konfiguracji:

Twierdzenie 3.3. *Istnieje funkcja $f : \mathbb{N} \to \mathbb{N}$ taka, że $\pi_l(T) \leqslant f(k)$ dla każdego drzewa T o szerokości ścieżkowej k.*

3.3 Ogólne grafy o ograniczonej szerokości ścieżkowej

Sekcję tę zaczniemy od konstrukcji grafu o szerokości ścieżkowej 2 i dowolnie dużej listowej liczbie Thuego. Niech $D_{n,k}$ będzie grafem powstałym ze ścieżki o długości $2n$ poprzez zastąpienie każdego nieparzystego wierzchołka grupą $\binom{kn}{k}$ wierzchołków, to jest:

$$V(D_{n,k}) = \{v_{2i} | i \in [n]\} \cup \{v^j_{2i-1} | i \in [n], j \in [\binom{kn}{k}]\}.$$

Wierzchołki są połączone krawędzia w $D_{n,k}$ wtedy i tylko wtedy, gdy ich indeksy dolne różnią się o dokładnie jeden (co w szczególności oznacza, że grupy wierzchołków nieparzystych tworzą zbiory niezależne). Udowodnimy teraz, że:

Twierdzenie 3.4. *Jeśli k i n są liczbami naturalnymi takimi, że $k > 1$ i $n > e^{k+2}$ to $\pi_l(D_{n,k}) > k$.*

Dowód. Na potrzeby dowodu wierzchołkiem uogólnionym V_i nazwijmy wierzchołek v_i dla i parzystych oraz zbiór wszystkich wierzchołków $v_{i,j}$ dla i nieparzystych. Wtedy dla kolorowania ϕ, przez $\phi(2i+1)$ będziemy rozumieć zbiór wszystkich kolorów wybranych dla wierzchołków w V_{2i+1}. Przyporządkujmy listy długości k do wierzchołków w taki sposób, że wierzchołki parzyste otrzymają parami rozłączne listy, natomiast wierzchołki nieparzyste o danym indeksie otrzymają wszystkie możliwe listy kolorów używanych przez listy wierzchołków parzystych. Dla wierzchołka v_i, gdzie $i = 2t$ połóżmy zatem $L_i = \{tk+1, tk+2, \ldots, tk+k\}$, a dla wierzchołków $v_{2i+1,1}, v_{2i+1,2}, \ldots, v_{2i+1,\binom{kn}{k}}$ niech $L_{2i+1,1}, \ldots$ będą wszystkimi k-elementowymi podzbiorami $\{1, 2, \ldots, kn\}$.

Przypuśćmy teraz, że dla takiego doboru list ϕ jest nierepetytywnym k-kolorowaniem $D_{n,k}$. By dojść do sprzeczności wystarczy rozważyć powtórzenia bloków nieparzystej długości. Zauważmy, że dla każdego segmentu $[D_i, D_{i+4j+2}]$ musi istnieć parzyste l takie, że $\phi(l) \notin \phi(l+2j+1)$ dla pewnego $j \in \mathbb{Z}$. Powiemy wtedy, że para $(l, l+2j+1)$ jest świadkiem dla segmentu $[D_i, D_{i+4j+2}]$. Oczywiście każdy segment musi mieć świadka, ponadto para $(l, l+2j+1)$ może być świadkiem jedynie dla zawierających ją segmentów długości $4j+2$, a zatem dla najwyżej $|2j+1|$ segmentów.

Graf $D_{n,k}$ zawiera $n-(4l+1)$ segmentów długości $4l+2$, które w sumie potrzebują co najmniej $\lceil \frac{n-(4l+1)}{2l+1} \rceil = \lceil \frac{n+1}{2l+1} - 2 \rceil$ różnych świadków. Sumując po wszystkich l otrzymujemy nierówność prawdziwą dla odpowiednio dużych n:

$$\Sigma_{l=1}^{\frac{n-2}{4}} \lceil \frac{n+1}{2l+1} - 2 \rceil \geqslant \frac{n}{2} \Sigma_{l=1}^{\frac{n}{4}} \frac{1}{l} - \frac{n}{2} \geqslant \frac{n}{2} \ln(\frac{n}{4}) - -\frac{n}{2} \geqslant \frac{n}{4} \ln(\frac{n}{4}). \qquad (6)$$

Zauważmy teraz, że uogólniony wierzchołek nieparzysty V_{2i+1} może należeć do najwyżej $k-1$ par typu $(l, 2i+1)$ będących świadkami, gdyż należenie do każdej z nich wiąże się z niemożliwością wykorzystania koloru $\phi(l)$ w żadnym z wierzchołków V_{2i+1}. Jeśli więc należałby on do k par, zabraniałyby one k różnych kolorów, zatem wierzchołek V_{2i+1} odpowiadający zabronionej k-tce nie mógłby zostać pokolorowany. Z równania 6 wynika, że uogólniony wierzchołek nieparzysty należy średnio do $\frac{1}{2} \ln(\frac{n}{4})$ par będących świadkami, co w połączeniu z założeniem $n > e^{k+2}$ daje nam pożądaną sprzeczność.

□

3.4 Drzewa o ograniczonej szerokości ścieżkowej

Reszta niniejszego rozdziału poświęcona będzie dowodowi twierdzenia o ograniczonej listowej liczbie chromatycznej drzew z ograniczoną szerokością ścieżkową. W całym dowodzie utrzymywana będzie konwencja, wedle której ko-

rzeń drzewa jest jego najniższym wierzchołkiem, a liście - najwyższymi. Metoda będzie bardzo podobna do wykorzystanej w dowodzie twierdzenia 2.4:

Twierdzenie 3.5. *Istnieje funkcja f taka, że jeśli T jest drzewem o szerokości ścieżkowej w to $\pi_l(T) < f(w)$.*

Ze względów technicznych zanim przejdziemy do dowodu uogólnimy pojęcie powtórzenia. *Powtórzeniem* długości n *z przerwą* t nazywać będziemy ciąg $(a_1, \ldots, a_n, b_1, \ldots, b_t, a_{n+1}, \ldots, a_{2n})$, gdzie $a_i = a_{i+n}$. Powtórzenie długości k nazywać będziemy w skrócie k-powtórzeniem. Nadużywając lekko notacji ścieżkę w grafie, w którym sekwencja kolorów tworzyć będzie powtórzenie również nazywać będziemy powtórzeniem. *Środkową krawędzią* powtórzenia długości k nazwiemy krawędź pomiędzy k-tym, a $k+1$-szym wierzchołkiem ścieżki. *Bliźniakiem* l-tego wierzchołka w powtórzeniu długości k nazwiemy wierzchołek o indeksie $(l+k) \mod 2k$.

W dowodzie korzystać będziemy z następującego lematu o dekompozycji drzew o ograniczonej szerokości ścieżkowej:

Lemat 3.6. *Jeśli T jest drzewem o szerokości ścieżkowej k, to istnieje zbiór P rozłącznych ścieżek należących do T taki, że:*

1. *Każdy wierzchołek T należy do dokładnie jednej ścieżki w P,*
2. *Metadrzewo $\mathbb{T}$ powstałe z T przez kontrakcję wszystkich ścieżek w P ma średnicę nie większą niż $2^{k+1} - 1$.*

Dowód. Dowód będzie przebiegać indukcyjnie ze względu na k. Przypadek $k = 0$ jest oczywisty, załóżmy więc, że $k > 0$ i teza zachodzi dla wszystkich $l < k$. Każdy graf o szerokości ścieżkowej k jest spójnym podgrafem grafu przedziałowego o wysokości $k+1$, weźmy więc dowolną reprezentację takiego nadgrafu dla T. Niech P_0 będzie ścieżką łączącą przedział zaczynający się najbardziej na lewo w tej reprezentacji z przedziałem kończącym się najbardziej na prawo (jest ona wyznaczona jednoznacznie, bo T jest drzewem). Po

usunięciu P_0 z reprezentacji, pozostaje las o szerokości ścieżkowej nie większej niż $k-1$. Z założenia indukcyjnego każde pozostałe drzewo posiada dekompozycję na ścieżki spełniające warunki twierdzenia tworzące metadrzewa o szerokości nie większej niż $2^k - 1$. Niech P będzie sumą tych ścieżek i P_0. Oczywiście ścieżki P są podziałem wierzchołków drzewa T, a średnica metadrzewa P jest nie większa niż $2^{k+1} - 1$. □

Ustalmy teraz drzewo T o szerokości ścieżkowej k wraz z dowolnym zanurzeniem na płaszczyznę, korzeniem oraz dekompozycją P o metadrzewie $\mathbb{T}$. Dodatkowo skierujmy każdą ścieżkę w P w prawą stronę. Pojedynczą ścieżkę dekompozycji nazywać będziemy ścieżką *pierwotną*, natomiast jej wierzchołek sąsiadujący ze ścieżką bliższą korzeniowi T - punktem *centralnym* (oczywiście ścieżka zawierająca korzeń nie ma punktu centralnego). Dla każdego wierzchołka $v \in T$, *poziomem* v będziemy nazywać odległość metawierzchołka zawierającego v do metawierzchołka zawierającego korzeń T w metadrzewie $\mathbb{T}$. Dla dowolnej ścieżki w T, jej fragment składający się z wierzchołków na najniższym osiąganym przez nią poziomie nazywać będziemy *dolną* częścią ścieżki. Scieżkę w T nazywać będziemy *stabilną* jeśli oba z jej końców są w jej dolnej części, *jednokierunkową* - jeśli leży tam tylko jeden z końców, oraz *dwukierunkową* - jeśli żaden z końców nie leży w dolnej części ścieżki. Jednokierunkowa ścieżka $(v_1, \ldots, v_l)$ jest skierowana *w lewo* jeśli (v_2, v_1) jest skierowaną krawędzią w ścieżce P zawierającej v_1, w przeciwnym wypadku taka ścieżka jest skierowana *w prawo* (nawet jeśli v_1 i v_2 należą do różnych ścieżek w P, definicje nie są zatem w pełni symetryczne). Dla ścieżek jedno- i dwukierunkowych wierzchołki należące do dolnej części ścieżki mające sąsiada nienależącego do dolnej częśći nazywać będziemy *punktami załamania*. Dla punktu v na poziomie k, *dziećmi* tego punktu nazwiemy wszystkie wierzchołki z poziomu $k+1$, które sąsiadują z v. Przez $v\uparrow$ oznaczać będziemy sumę wszystkich potomków v wraz z v.

Zanim przejdziemy do konstrukcji nierepetytywnego kolorowania T skró-

cimy listy wykluczając jednocześnie następujące szczególne typy powtórzeń:

(1) k-powtórzenia z przerwą t, gdzie $t < k$, należące w całości do jednej ścieżki pierwotnej,

(2) jednostronne k-powtórzenia skierowane w prawo z dowolnie dużą przerwą takie, że przecięcie przerwy i dolnej części powtórzenia jest nie dłuższe niż k,

(3) jednostronne k-powtórzenia skierowane w lewo z dowolnie dużą przerwą takie, że przecięcie przerwy i dolnej części powtórzenia jest nie dłuższe niż k.

Opis stopniowego skracania list opiszemy w trzech lematach poświęconych trzem kolejnym typom powtórzeń. Skracanie list w drzewie można rozważać jako kolorowanie drzewa krótszymi listami, każde takie kolorowanie nazywać będziemy *L-kolorowaniem*. We wszystkich dowodach wszystkie skrócone listy będą tej samej długości. Ścieżkę pokolorowaną listami nazywać będziemy *L-powtórzeniem* danego typu, jeśli z list da się wybrać kolory w taki sposób, by powstało powtórzenie tego typu.

Ze względów technicznych będziemy potrzebować następującej obserwacji:

Lemat 3.7. *Jeśli $x_1, \ldots, x_n$ jest ciągiem nieujemnych liczby rzeczywistych takich, że $\Sigma_{i=1}^{n} x_i = M$ to $\Pi_{i=1}^{n} x_i \leqslant 4^M$.*

Dowód. Z nierówności Cauchy'ego wynika, że iloczyn n liczb o stałej sumie jest maksymalizowany, gdy liczby te są równe. Interesuje nas zatem maksymalna wartość wyrażenia x^n przy założeniu, że $nx = M$. Zauważmy, że dla $x > 4$ mamy $(\frac{x}{2})^2 > x$, a zatem wybranie $x' = \frac{x}{2}$ i $n' = 2n$ zwiększy nasze wyrażenie. Maksimum jest więc osiągane dla x-ów mniejszych niż 4, co implikuje nierówność $x^n < 4^n < 4^M$. □

Lemat 3.8. *Jeśli T jest drzewem o szerokości ścieżkowej k z listami wielkości $N = 64n^2 + n - 1$ przyporządkowaymi do wierzchołków to istnieje L-kolorowanie T listami wielkości n w taki sposób, by nie istniało L-powtórzenie typu* (1).

Dowód. Niech T będzie drzewem o szerokości drzewowej k, metadrzewie $\mathbb{T}$ i ze zbiorem list $\{L_v\}_{v \in V(T)}$ o wielkościach równych $64n^2+n-1$ dla każdego v. Załóżmy, że nie jest możliwy wybór podlist wielkości n w taki sposób, by nie istniała L-powtórzenie typu (1). Z definicji każde powtórzenie typu (1) zawarte jest w jednej ścieżce pierwotnej, a zatem istnieje pewna ścieżka $P \in \mathbb{P}$, dla której nie istnieje odpowiedni wybór podlist. Rozważmy randomizowany algorytm (Algorytm 2 poniżej) próbujący wybrać podlisty w taki sposób, by nie stworzyć L-powtórzeń typu (1).

Algorithm 2: Unikanie powtórzeń typu (1)

```
1  wejście: S : ℕ → [(N choose n)]
2  i ← 1, j ← 1, C ← ∅
3  while Ścieżka nie jest w całości L-pokolorowana do
4      C(v_j) ← podlista L o indeksie S(i)
5      i++
6      j++
7      if Istnieje instancja R L-powtórzenia kończąca się na v_j then
8          Niech W_R będzie powtórzoną częścią R zawierającą v_j
9          for v ∈ W_R do
10             usuń warotść C(v)
11         j ← indeks pierwszej pozycji w W_R
12 return C
```

Zauważmy, że dzięki założeniu o nieistnieniu poprawnego L-kolorowania, algorytm nigdy się nie zatrzyma. Ustalmy teraz pewną sekwencję wejściową

S i prześledźmy pierwsze M kroków działania algorytmu. Tak jak poprzednio, postaramy się opisać działanie algorytmu za pomocą struktur, których ilość oszacujemy później za pomocą metod kombinatoryki analitycznej. Tym razem strukturą będzie czwórka (P, F, G, R), gdzie:

- $P = (p_1, \ldots, p_M)$ jest ciągiem liczb, gdzie p_i jest jest ilością wierzchołków z nadanymi kolorami (mniejszymi listami) po i krokach,
- F jest funkcją częściową C po M krokach algorytmu,
- $G = (g_1, \ldots, g_s)$ jest ciągiem liczb takich, że g_i jest długością przerwy w i-tej wycofanego powtórzenia typu (1),
- $R = (r_1, \ldots, r_s)$ jest ciągiem sekwencji liczb, gdzie $r_i = (l_1, \ldots, l_k)$ odpowiada i-temu powtórzeniu. Jeśli i-te powtórzenie pojawiło się na ścieżce $(v_1, \ldots, v_k, b_1, \ldots, b_t, v_{k+1}, \ldots, v_{2k})$, gdzie t jest długością przerwy, a listy je tworzące to $L_1, \ldots, L_k, B_1, \ldots, B_t, L_{k+1}, \ldots, L_{2k}$ to l_i jest indeksem listy L_{k+i} spośród wszystkich list możliwych do wybrania na wierzchołku v_i.

Teraz przystąpimy do dowodów kompresji i bezstratności tego kodowania, analogicznie jak miało to miejsce w poprzednim rozdziale.

Kompresja

1. Zacznijmy od oszacowania ilości wszystkich możliwych P. Ciąg $(p_1, \ldots, p_M)$ bijektywnie przetransformujmy w ciąg różnic $(1, p_2 - p_1, \ldots, p_M - p_{M-1})$ tak, że wszystkie różnice należeć będą do zbioru $\{1, 0, -1, -2, \ldots\}$. Następnie każdy niedodatni element $-k$ tego ciągu zamieńmy w podciąg $(1, -1, -1, \ldots)$ długości k, łatwo sprawdzić, że ta operacja również jest bijektywna. Ciąg wynikowy jest ciągiem zerojedynkowym o długości nie większej niż $2M$, zatem ilość takich ciągów jest rzędu $O(4^M)$, więc i ilość P jest takiego rzędu, jako że obie transformacje były bijektywne.

2. Liczba możliwych F może być traktowana jak stała, gdyż nie rośnie wraz z M.

3. Suma elementów G jest nie większa niż suma długości wszystkich wycofanych L-powtórzeń, która jest nie większa niż M, zatem ilość możliwych ciągów G jest mniejsza niż ilość ciągów $M+1$ liczb nieujemnych sumujących się do M, czyli $\binom{2M}{M} = O(4^M)$.

4. By oszacować liczbę możliwych R przypomnijmy, że każde l_j występujące w r_i będącym elementem R jest indeksem pewnej n-podlisty pośród wszystkich n-podlist listy rozmiaru N mających niepuste przecięcie z pewną jej ustaloną podlistą rozmiaru n. Indeks ten nie może być większy niż $n\binom{N-1}{n-1}$. Suma długości sekwencji składających się na R jest nie większa niż M, a każdy ciąg długości M może być rozdzielony na spójne podciągi na 2^{M-1} sposobów. Aby zakodować R należy osobno zakodować wyrazy należące do wszystkich sekwencji oraz sposób podziału na sekwencje, co w połączeniu z poprzednimi obserwacjami daje pożądane oszacowanie postaci $O((2n\binom{N-1}{n-1})^M)$.

Mnożąc uzyskane szacowania dostajemy oszacowanie na ilość możliwych czwórek (P, F, G, R) opisujących konkretny przebieg algorytmu o długości M:

$$O\Big(\Big(2^5 n\Big(\binom{N-1}{n-1}\Big)\Big)^M\Big) = O\Big(\Big(2^5 n \frac{n}{N-n+1}\binom{N}{n}\Big)^M\Big) = O\Big(\binom{N}{n}^M\Big).$$

Widać zatem, że dla wystarczająco dużego M liczba możliwych opisów działania algorytmu jest mniejsza niż liczba początkowych segmentów S o długości M. By uzyskać sprzeczność wystarczy teraz pokazać, że dwa różne segmenty początkowe S generują zawsze różne opisy.

Bezstratność opisu

Udowodnimy, że da się zrekonstruować pierwsze M elementów ciągu S na podstawie czwórki (P, F, G, R) powstałej po M krokach działania algorytmu.

Argument będzie iteracyjny - dla (P, F, G, R) zrekonstruujemy $S(M)$ oraz (P', F', G', R') - czwórkę powstałą po $M-1$ krokach działania algorytmu, a następnie powtarzając rozumowanie - całą początkową sekwencję S. Jeśli $p_M = p_{M-1} + 1$ to żadne L-powtórzenie nie zostało wycofane w ostatnim kroku. Zatem $S(M)$ to po prostu indeks ostatniej podlisty F, P' i F' są o jeden element krótsze, a $G' = G$ i $R' = R$. Jeśli $p_M = p_{M-1} - k + 1$, gdzie $k > 0$, to w ostatnim kroku została wykonana retrakcja k elementów tworzących l-powtórzenie. Ostatni element G to długość luki t dla tego powtórzenia.

Niech $(\alpha_1, \ldots, \alpha_r)$ będzie ostatnim elementem R, a $a_1, \ldots, a_k, b_1, \ldots, b_t$ ostatnimi elementami F o przypisanych podlistach, oraz niech $A_1, \ldots, A_k$ będą podlistami przypisanymi do $a_1, \ldots, a_k$ (informacje o tych podlistach są zapisane w F). Ponadto niech $a_{k+1}, \ldots, a_{2k}$ będą wierzchołkami następującymi po b_t na ścieżce - są to wierzchołki, którym przed retrakcją były przypisane podlisty. Rozważmy wierzchołek a_{k+i}. Znamy jego listę $L(a_{k+i})$, znamy podlistę A_i przypisaną do a_i i wiemy, że lista A_{k+1} przypisana wcześniej do a_{k+i} ma indeks α_i spośród wszystkich n-podlist $L(a_{k+i})$ mających niepuste przecięcie z A_i. Pozwala nam to zrekonstuować wszystkie podlisty przypisane wierzchołkom przed retrakcją - F''. Tak jak w poprzednim przypadku, $S(M)$ to indeks ostatniej podlisty F'', natomiast (P', F', G', R') są o jeden element krótsze niż (P, F'', G, R). □

Używając bardzo podobnych argumentów w kolejnych lematach pokażemy, że jeśli początkowe listy są wystarczająco długie to mogą zostać skrócone w taki sposób, by każde kolorowanie unikało powtórzeń typu (2) oraz (3).

Lemat 3.9. *Jeśli T jest drzewem o szerokości ścieżkowej k i listach rozmiaru $N = 64n^2 + n - 1$ oraz bez powtórzeń typu (1) to da się wybrać podlisty rozmiaru n dla wszystkich wierzchołków tak, by nie istniało kolorowanie z tych podlist zawierające powtórzenie typu (2) o dolnej części w ścieżce pierwotnej*

zawierającej korzeń T.

Dowód. W obrębie dowodu powtórzenie typu (2) z dolną częścią w ścieżce zawierającej korzeń T nazywać będziemy powtórzeniem typu $(2')$. Dla częściowego L-kolorowania C i wierzchołka v niech $d(C, v)$ będzie deterministycznie skonstruowanym rozszerzeniem C o wszystkie wierzchołki należące do $v\uparrow$, które unika powtórzeń typu $(2')$. Oczywiście takie rozszerzenie nie musi zawsze istnieć, więc d jest funkcją częściową. Jeśli istnieje wiele takich rozszerzeń, wybieramy pierwsze zgodnie z dowolnym ustalonym porządkiem. Naturalnie rozszerzamy d do funkcji, która działa na zbiorach wierzchołków, a nie tylko na pojedynczych wierzchołkach.

Dowód będzie polegał na analizie algorytmu 3 kolorującego drzewo i unikającego powtórzeń typu $(2')$. W analizie algorytmu, wartości $C(v)$ przyporządkowane w linii 5 nazywać będziemy przyporządkowanymi *bezpośrednio*. Algorytm korzysta ze zdefiniowanej później funkcji `Next` oraz dodatkowych oznaczeń - dla wierzchołka v ze ścieżki pierwotnej $\mathbb{P}$ przez `right`(v) oznaczać będziemy wierzchołek na prawo od v, natomiast przez `left`(v) - na lewo od v.

Załóżmy, że nie istnieje L-kolorowanie drzewa T unikające powtórzeń typu $(2')$. Podobnie jak w przypadku ostatniego dowodu, prześledzimy teraz pierwsze M kroków algorytmu (który, znów podobnie jak w ostatnim przypadku, nigdy się nie zatrzyma). Zauważmy, że w każdym momencie trwania algorytmu wierzchołki pokolorowane bezpośrednio tworzą ścieżkę, a ponadto gdy w trakcie trwania procedury powstaje prekolorowanie typu $(2')$, aktualnie kolorowany wierzchołek musi być końcem ścieżki zawierającej to powtórzenie. Z faktu, że kolorowania poddrzew ukorzenionych w bezpośrednio pokolorowanych wierzchołkach są wybierane deterministycznie wynika ponadto, że częściowe kolorowanie całego drzewa jest całkowicie zdeterminowane przez bezpośrednio pokolorowane wierzchołki. Co więcej - i-ty wierzchołek bezpośrednio pokolorowanej ścieżki $(v_1, \ldots, v_r)$ jest zdeterminowany przez podlisty

Algorithm 3: Eliminacja L-powtórzeń typu (2')

input: $S : \mathbb{N} \to [\binom{N}{n}]$
`i` $\leftarrow 1$, $C \leftarrow \emptyset$
$v \leftarrow$ korzeń T
while *T nie jest całkowicie l-pokolorowane* **do**
    $C(v) \leftarrow$ podlista L_v długości n o indeksie $S(i)$
    `i++`
    if *istnieje L-powtórzenie P typu (2')* **then**
        Niech P' będzie powtórzoną częścią P
        for $v' \in P'\uparrow$ **do**
            usuń wartość $C(v')$
        $v \leftarrow$ punkt P' najbliższy korzeniowi
    else
        $(C, v) \leftarrow \text{Next}(C, v)$

$(v_1, \ldots, v_{i-1})$ (v_1 jest korzeniem).

W ogólności funkcja `Next` jest niezdefiniowana dla pewnych argumentów (np. gdy v jest pierwszym wierzchołkiem meta-ścieżki i nie ma dzieci). Mimo to, jako że założyliśmy niemożność pokolorowania drzewa bez powtórzeń, a funkcja `Next` zawsze wybiera poddrzewo dla którego aktualne kolorowanie C nie może być rozszerzone, pewne L-powtórzenie musi się pojawić przed osiągnięciem wierzchołka, dla którego funkcja jest niezdefiniowana.

Przeanalizujmy pierwsze M kroków algorytmu działającego na sekwencji S. Informacje dotyczące przebiegu znów zapisywać będziemy jako czwórkę (P, F, G, R), gdzie F i R zdefiniowane będą tak jak poprzednio, w ciągu $P = \{p_1, \ldots, p_M\}$ element p_i oznacza ilość wierzchołków **bezpośrednio** pokolorowanych po i krokach, natomiast $G = (g_1, \ldots, g_t)$ jest ciągiem, gdzie g_i to ilość wierzchołków i-tego powtórzenia, które należą do przecięcia luki i

Algorithm 4: Funkcja Next

input: C częściowe l-kolorowanie drzewa
input: v wierzchołek drzewa
if $d(C, v)$ *nie jest zdefiniowane* **then**
    Niech $(v_1, \ldots, v_k)$ będzie numeracją dzieci v
    Niech j będzie największą liczbą, dla której $d(C, \{v_1, \ldots, v_j\})$ jest zdefiniowane
    $v' \leftarrow v_{j+1}$
    $C' \leftarrow d(C, v_1, \ldots, v_j)$
else
    C' $\leftarrow d(C, v)$
    if v *jest centralnym punktem ścieżki pierwotnej* $\mathbb{P}$ **then**
        Niech B będzie zbiorem wierzchołków na prawo od v w $\mathbb{P}$ razem z v
        if $d(C, B)$ *istnieje* **then**
            $v' \leftarrow \text{left}(v)$
            $C' \leftarrow d(C, B)$
        else
            $v' \leftarrow \text{right}(v)$
    else if v *jest na lewo od centralnego punktu* $\mathbb{P}$ **then**
        $v' \leftarrow \text{left}(v)$
    else if v *jest na prawo od centralnego punktu* $\mathbb{P}$ **then**
        $v' \leftarrow \text{right}(v)$
return (C', v')

dolnej części powtórzenia. Z definicji, g_i jest nie większe niż długość powtórzenia.

Zauważmy, że wierzchołki na ścieżce będącej korzeniem meta-drzewa mogą

zostać pokolorowane jedynie bezpośrednio oraz że ostatni wierzchołek tworzący wycofywane powtórzenie również jest kolorowany bezpośrednio, a zatem każde powtórzenie składa się jedynie z wierzchołków bezpośrednio pokolorowanych.

Kompresja Przeanalizujmy asymptotyczną liczbę czwórek opisujących pierwsze M kroków algorytmu, gdzie M zmierza do nieskończoności. Oszacowanie P, F, G oraz R jest identyczne jak w dowodzie lematu 3.9, zatem liczba możliwych czwórek wynosi:

$$o\left(\left(2^5 n\binom{N-1}{n-1}\right)^M\right) = o\left(\left(2^5 n\frac{n}{N-n+1}\binom{N}{n}\right)^M\right) = o\left(\binom{N}{n}^M\right).$$

Podobnie jak poprzednio wnioskujemy zatem, że dla wystarczająco dużego M ilość możliwych opisów jest mniejsza niż ilość początkowych segmentów S, co prowadzi do sprzeczności.

Bezstratność Znów tak jak w dowodzie poprzedniego lematu zastosujemy rozumowanie iteracyjne by z czwórki (P, F, G, R) otrzymać elementy ciągu S.

Jeśli $p_M = p_{M-1} + 1$, to żadne powtórzenie nie zostało wycofane w poprzednim kroku. Z każdego częściowego kolorowania, w szczególności z F, jesteśmy w stanie wydedukować, które wartości zostały przyporządkowane bezpośrednio - tworzą one ścieżkę $(v_1, \ldots, v_m)$, gdzie v_1 jest pierwszym wierzchołkiem ścieżki będącej korzeniem metadrzewa. Wtedy $S(M)$ jest indeksem podlisty przyporządkowanej v_m w F, $R' = R, G' = G$, P' to P bez ostatniego elementu, natomiast F' jest częściowym kolorowaniem determinowanym przez podlisty bezpośrednio przyporządkowane $(v_1, \ldots, v_{m-1})$ w F.

Jeśli $p_M = p_{M-1} - k + 1$ i $k > 0$, to po ostatnim kroku zostało wycofane L-powtórzenie o długości k. Niech g będzie ostatnim elementem G, a $r = (\alpha_1, \ldots, \alpha_k)$ - ostatnim elementem R. Zrekonstruujemy kolorowanie zawierające powtórzenie w k fazach. Ścieżka z przyporządkowanymi podlistani

jednoznacznie wyznacza następny wierzchołek v', któremu zostanie bezpośrednio przyporządkowana podlista. Jest to również ostatni wierzchołek, którego podlista została usunięta podczas retrakcji. Wiemy również, że podlista do niego przyporządkowana miała niepuste przecięcie z podlistą wierzchołka w należącego do drugiej powtórzenia, którego pozycję (a zatem i jego podlistę A_w) potrafimy odtworzyć znając g. W połączeniu z α_1 pozwala nam to odtworzyć podlistę przyporządkowaną wcześniej do v'. Iterując argument k-krotnie otrzymujemy F''. By otrzymać F' wystarczy usunąć ostatnio przyporządkowaną podlistę F'', natomiast P', R' i G' równe są P, R i G z usuniętymi ostatnimi elementami. □

Lemat 3.10. *Dla każdego k istnieje N_k takie, że jeśli T jest drzewem o szerokości ścieżkowej T z listami rozmiaru N_k w każdym wierzchołku bez powtórzeń typu* (1)*, to możliwe jest wybranie podlist rozmiaru n, które unikać będą powtórzeń typu* (2) *i* (3)*.*

Dowód. By pozbyć się powtórzenia typu (2) o dolnej części na danej ścieżce należącej do meta-drzewa wystarczy zastosować procedurę z lematu 3.9. Stosując ją dla wszystkich ścieżek pierwotnych usuniemy wszystkie tego typu powtórzenia. Zauważmy, że każda ścieżka pierwotna ma najwyżej $k-1$ przodków w drzewie, a zatem każda lista zostanie skrócona najwyżej k razy. By pozbyć się powtórzeń typu (3) wystarczy odwrócić kierunek wszystkich ścieżek pierwotnych i ponownie zastosować poprzednie rozumowanie. □

Lemat 3.11. *Jeśli T jest drzewem o szerokości ścieżkowej k i listach o rozmiarze 2^{k+1}, które nie zawiera L-powtórzeń typów* $(1), (2), (3)$*, to T może zostać nierepetytywnie pokolorowane.*

Dowód. Uporządkujmy wierzchołki drzewa rosnąco ze względu na odległość od korzenia w meta-drzewie (rozstrzygając remisy w dowolny sposób). Będziemy kolorować wierzchołki zachłannie w taki sposób, że za każdym razem

gdy wierzchołkowi v będzie przyporządkowywany kolor c, kolor ten będzie usuwany z list wierzchołków należących do $v\uparrow$. Jako że głębokość skontraktowanego drzewa jest mniejsza niż długość list, żadna lista nigdy nie będzie pusta.

Załóżmy teraz, że w tak pokolorowanym drzewie pojawiło się powtórzenie. Jeśli jest ono na ścieżce jednokierunkowej, jego środkowa krawędź musi być na jego dolnej części (co wynika z procedury usuwania kolorów z list) - łatwo wtedy sprawdzić, że powtórzenie takie byłoby typu (1), (2) lub (3), co daje sprzeczność.

Jako, że powtórzenia należące w całości do ścieżek pierwotnych już wykluczyliśmy, jedyną pozostającą możliwością jest powtórzenie dwustronne - takie, którego żaden z końców nie leży na najniższym osiąganym przez nie poziomie. Jednak tak jak w poprzednim przypadku, jego krawędź środkowa musi leżeć w jego dolnej częsci. Bez straty ogólności przyjmijmy, że krawędź środkowa jest nie odleglejsza od lewego punktu schodzącego, niż od prawego punktu schodzącego - w tej sytuacji powtórzenie typu (2) jest podścieżką powtórzenia dwustronnego, co daje sprzeczność. □

Tezę twierdzenia 3.5 uzyskujemy stosując lematy 3.9, 3.10 do pozbycia się powtórzeń typu (1), (2) oraz (3), a następnie kolorując wierzchołki drzewa przy pomocy lematu 3.11.

Literatura

[1] Alon, N., Grytczuk, J., Hałuszczak, M., Riordan, O. *Nonrepetitive colorings of graphs*, Random Structures Algorithms 21 (2002), 336–346.

[2] Bean, D.R., Ehrenfeucht, A., McNulty, G. *Avoidable patterns in strings of symbols*, Pacific Journal of Mathematics 85 (1979), 261–294.

[3] Blanchet-Sadri, F., Merca¸s, R., Simmons, S., Weissenstein, E. *Avoidable binary patterns in partial words*, Acta Informatica 48 (2011), 25–41.

[4] Cassaigne, J. *Unavoidable binary patterns*, Acta Informatica 30 (1993), 385–395.

[5] Crochemore, M., Ilie, L., Rytter, W. *Repetitions in strings: Algorithms and combinatorics*, Theoretical Computer Science 410 (2009), 5227–5235.

[6] Currie, J.D. *Pattern avoidance: themes and variations*, Theoretical Computer Science 339 (2005), 7-18.

[7] Dujmović, V., Joret, G., Kozik, J., Wood, D.R. *Nonrepetitive colouring via entropy compression*, Combinatorica 36 (2016), 661–686.

[8] Gągol, A. *Pattern avoidance in partial words over a ternary alphabet*, Annales UMCS 69 (2015), 73-82.

[9] Gągol, A., Joret, G., Kozik, J., Micek, P.*Pathwidth and Nonrepetitive List Coloring*, Electr. J. Comb. 23 (2016), 4-40.

[10] Flajolet, P., Sedgewick, R. *Analytic Combinatorics*, Cambridge University Press (2009), ISBN 978-0-521-89806-5.

[11] Fiorenzi, F., Ochem, P., Ossona de Mendez, P., Zhu, X. *Thue choosability of trees*, Discrete App. Math. 17 (2011), 2045-2049.

[12] Grytczuk, J., Kozik, J., Micek, P. *A new approach to nonrepetitive sequences*, Random Structures and Algorithms 42 (2013), 214-225.

[13] Grytczuk, J. *Nonrepetitive graph coloring*, Graph Theory in Paris, Trends in Mathematics (2007), 209–218.

[14] Grytczuk. J. *Pattern avoidance on graphs*, Discrete Math 11-12 (2007), 1341–1346.

[15] Harant J. and Jendrol', S. *Nonrepetitive vertex colorings of graphs*, Discrete Math. 2 (2012), 374–380.

[16] Kündgen, A. and Pelsmajer J. M., *Nonrepetitive colorings of graphs of bounded tree-width*, Discrete Math. 19 (2008), 4473–4478.

[17] Krieger, D.,Ochem, P., Rampersad, N., Shallit, J. *Avoiding Approximate Squares*, Lecture Notes in Computer Science 4588 (2007), 278-289.

[18] Lothaire, M. *Algebraic Combinatorics on Words*, Cambridge University Press, Cambridge (2002).

[19] Erdős, P., Lovász, L. *Problems and results on 3-chromatic hypergraphs and some related questions*, Infinite and Finite Sets (to Paul Erdős on his 60th birthday). II. North-Holland. (1975) 609–627.

[20] Manea, F., Mercaş, R. *Freeness of partial words*, Theoretical Computer Science 389 (2007), 265–277.

[21] Moser, R.A., Tardos, G. *A constructive proof of the general lovasz local lemma*, Journal of ACM 57 (2010), 1-15.

[22] Roth, P. *Every binary pattern of length six is avoidable on the two-letter alphabet*, Acta Informatica 29 (1992), 95–106.

[23] Schmidt, U. *Avoidable patterns on two letters*, Theoretical Computer Science 63 (1989), 1–17.

[24] Szele, T. *Kombinatorikai vizsgálatok az irányitott teljes gráffal*, Kapcsolatban, Mt. Fiz. Lapok 50 (1943), 223-256.

[25] Thue, A. *Über unendliche Zeichenreihen*, Norske Vid. Selsk. Skr. I, Mat. Nat. Kl. Christiana 7 (1906), 1–22.

[26] Thue, A. *Über die gegenseitige Lage gleicher Teile gewisser Zeichenreihen*, Norske Vid. Selsk. Skr. I, Mat. Nat. Kl. Christiana 1 (1912), 1–67.

[27] Zimin, A.I. *Blocking sets of terms*, Mathematics of the USSR-Sbornik 47 (1984), 353–364.