

Mikołaj Bojańczyk

Urodzony: 8 czerwca 1977

e-mail: bojan@mimuw.edu.pl

Adiunkt w Instytucie Informatyki UW



## Zainteresowania badawcze

Większość moich badań dotyczy automatów skończonych na drzewach. Automat taki czyta drzewo skończone binarne, rozpoczynając od liści, a kończąc na korzeniu (rzecz jasna są też inne warianty). Ma on skończony zbiór stanów  $Q$  oraz zbiór reguł postaci: “jeśli wyliczyłem lewe poddrzewo do stanu  $p \in Q$ , a prawo poddrzewo do stanu  $q \in Q$ , zaś etykietą korzenia jest  $a$ , to całe drzewo wyliczam stanu  $r \in Q$ ”. Automatem takim można sprawdzić na przykład, czy w drzewie jest parzysta ilość wystąpień etykiety  $a$ , ale nie można sprawdzić, że w drzewie etykiety  $a$  oraz  $b$  występują tyle samo razy.

Poniżej przedstawione są konkretne zagadnienia, którymi się zajmowałem.

*Automaty ścieżkowe.* Automat na drzewach, tak jak został przedstawiony powyżej, jest maszyną równoległą: dla przetworzenia całego drzewa, musi przetwarzać jednocześnie jego poddrzewa (począwszy od wszystkich liści). Istnieje też naturalny model sekwencyjny, w którym automat przechodzi drzewo wierzchołek po wierzchołku. Reguły w tym modelu są postaci: “jeśli jestem w wierzchołku o etykiecie  $a$  i mam stan  $q \in Q$ , to ruszam się w kierunku  $d \in \{\nearrow, \nwarrow, \searrow, \swarrow\}$  i przechodzę do stanu  $r \in Q$ ”. Automaty takie zostały wprowadzone na początku lat siedemdziesiątych, niemniej przez trzydzieści lat nie wiadomo było, czy każdy (równoległy) automat na drzewach można równoważnie opisać za pomocą automatu ścieżkowego. Wspólnie z Thomasem Colcombetem udzieliliśmy negatywnej odpowiedzi na to pytanie [7]; pokazaliśmy też, że automatów ścieżkowych nie można determinizować [6].

*Algebra drzew.* Automaty na skończone słowach można równoważnie opisywać jako półgrupy skończone. Obserwacja ta leży u podstaw licznych związków między teorią języków formalnych a algebrą nieprzemianą. Na przykład twierdzenie Schützenbergera mówi, że język regularny słów można opisać wyrażeniem bez gwiazdki  $L^*$  wtedy i tylko wtedy, gdy żadna nietrywialna grupa nie dzieli półgrupy syntaktycznej języka. Wspólnie z Igozem Walukiewiczem poszukujemy podobnych wyników dla języków drzew [12].

*Alfabety nieskończone.* Zazwyczaj przymuje się, że zbiór liter  $a$ , które mogą etykietować wierzchołki drzewa, jest skończony. Niemniej w rzeczywistych zastosowaniach zbiór ten może być potencjalnie nieskończony. Jeżeli na przykład drzewo jest dokumentem XML, to etykieta może zawierać liczbę, albo nawet cały długi tekst. Prace [9,10] przedstawiają modele automatowe, w których alfabet jest nieskończony. Automaty takie muszą być bardzo ograniczone, gdyż

nawet najprostsze własności dotyczące alfabetów nieskończonych są co najmniej tak trudne, jak problem osiągalności w sieciach Petriego.

## Wykształcenie

1991-1995. Pierwsze Społeczne Liceum Ogólnokształcące w Warszawie.

1995-2000. Studia informatyczne na Uniwersytecie Warszawskim. Praca magisterska: “Problem skończonego modelu dla dwukierunkowych automatów alternujących” pod opieką Igora Walukiewicza.

2000-2004. Studia doktoranckie na Uniwersytecie Warszawskim. Doktorat: “Rozstrzygalne własności języków drzew”, pod opieką Igora Walukiewicza.

## Publikacje

[1] M. Bojańczyk. *The finite graph problem for two-way alternating automata*, Foundation of Software Science and Computation Structures, volume 2030 of Lecture Notes in Computer Science, pages 88103, 2001.

[2] M. Bojańczyk. *Two-way alternating automata and finite models*. International Colloquium on Automata, Languages and Programming, volume 2380 of Lecture Notes in Computer Science, pages 833844, 2002.

[3] M. Bojańczyk. *1-bounded TWA cannot be determinized*. Foundations of Software Technology and Theoretical Computer Science, volume 2914 of Lecture Notes in Computer Science, pages 6273. Springer, 2003.

[4] M. Bojańczyk. *The finite graph problem for two-way alternating automata*. Theoretical Computer Science, 298(3):511528, 2003.

[5] M. Bojańczyk. *A bounding quantifier*. Computer Science Logic, volume 3210 of Lecture Notes in Computer Science, pages 4155, 2004.

[6] M. Bojańczyk and T. Colcombet. *Tree-walking automata cannot be determinized*. International Colloquium on Automata, Languages and Programming, volume 3142 of Lecture Notes in Computer Science, pages 246–256, 2004.

[7] M. Bojańczyk and T. Colcombet. *Tree-walking automata do not recognize all regular languages*. ACM Symposium on the Theory of Computing, pages 234243, 2005.

[8] M. Bojańczyk and T. Colcombet. *Omega-regular expressions with bounds*. Logic in Computer Science, 2006. To appear.

[9] M. Bojańczyk, C. David, A. Muscholl, T. Schwentick, and L. Segoufin. *Two-variable logic on data trees and XML reasoning*. Principles of Database Systems, 2006. To appear.

[10] M. Bojańczyk, C. David, A. Muscholl, T. Schwentick, and L. Segoufin. *Two-variable logic on words with data*. Logic in Computer Science, 2006. To appear.

[11] M. Bojańczyk, M. Samuelides, T. Schwentick, and L. Segoufin. *The expressive power of pebble tree automata* International Colloquium on Automata, Languages and Programming, 2006. to appear.

[12] M. Bojańczyk and I. Walukiewicz. *Characterizing EF and EX tree logics*. International Conference on Concurrency Theory, CONCUR, volume 3170 of Lecture Notes in Computer Science, pages 131145, 2004.