Taylor J. Smith

Research Statement

Biographical Sketch. As a PhD candidate at Queen's University, my research work belongs to the broad category of theoretical computer science. I work primarily in automata theory and formal language theory, with a focus on the descriptional complexity and computational complexity of automaton models and language operations. I have also published work relating to bio-inspired language operations and DNA computing, combinatorics on words, and coding theory.

My research work has received accolades such as the Sheng Yu Award for Best Paper at the 24th International Conference on Implementation and Application of Automata (CIAA 2019). The award-winning paper [4] was selected from a program of 17 papers, and an invited extended version of the paper is set to appear in a special issue of the journal *Theoretical Computer Science*. I have also received full funding throughout my doctoral studies from the Government of Ontario as a recipient of the Ontario Graduate Scholarship; my scholarship was one of six awarded in the Queen's School of Computing, with a total value of $60 000.

Research Summary. Every computer scientist has heard of a finite automaton. But not nearly as many have heard of a "two-dimensional automaton". Unconventional computational models such as this abound in computer science, and their utility in related (and sometimes unrelated) areas, despite offering fruitful avenues for further research, frequently goes unnoticed. The intent of my work is to change that.

An increasing interest in applications of formal languages, such as pattern recognition and image processing, led to the formalization of two-dimensional data as a matrix of symbols. In the real world, this formalization is akin to representing the pixels of an image by a tuple of numeric values. In response, researchers similarly extended the study of automata theory from one dimension to two. A two-dimensional automaton is an automaton that takes a matrix of symbols as input and moves its input head through the word in four directions.

My long-term goals as a researcher are twofold: to investigate the complexity and utility of unconventional computational models, and to discover the applicability of such models to other areas of computer science. In the early days of my master's degree, my supervisor shared with me a recent paper on "bi-periodic infinite pictures", and its contents inspired me so much that the problems I solved formed the basis for my master's thesis work. I found the structure and versatility of two-dimensional words appealing, and was curious about the kinds of abstractions we can achieve with such a construction. I later shifted from formal languages to automata theory, where I became fascinated by the stark contrast between two-dimensional automata and the classical finite automaton. Simply adding another dimension makes the model Turing-equivalent and, thus, difficult to reason about in a meaningful way. In response, I wanted to explore the "in-between": if we restrict the abilities of a two-dimensional automaton, what is the degree of computational power we obtain, and how much power do we retain with this model compared to a classical finite automaton?

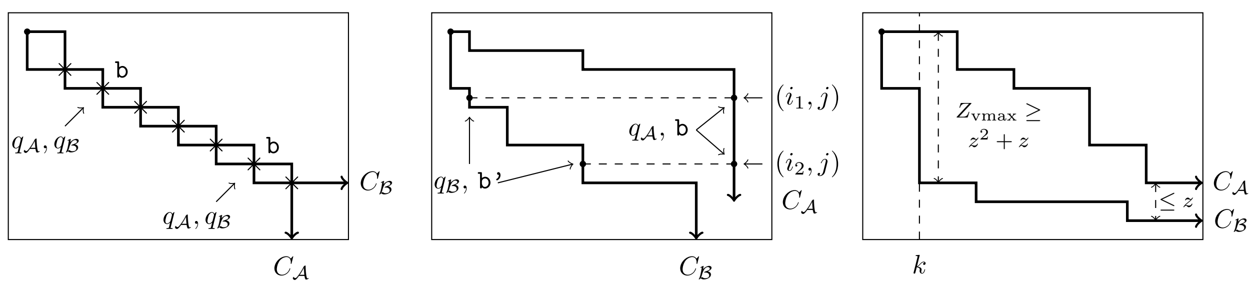
My work explores the decidability properties, closure properties, and complexity measures of variations of the two-dimensional automaton model, with a specific focus on variants where the input head of the automaton cannot move upward ("three-way" automata) or cannot move either upward or leftward ("two-way" automata). These models can be viewed as refinements positioned between the classical finite automaton and the two-dimensional automaton. Establishing the exact recognition and decidability power of these intermediate models allows us to select the most appropriate computational model for a given application.

As part of my PhD comprehensive examination, I wrote a survey article [3] with a dedicated focus on the basic models of deterministic and nondeterministic two-dimensional automata, their similarities and connections to one-dimensional automata, and the differences between two-dimensional automata and the three-way variant. I viewed the literature through three lenses: language recognition capabilities, closure and decidability properties, and complexity bounds. This is the first survey to analyze previous work from this perspective, and I plan to expand the article for submission to a venue such as *ACM Computing Surveys*.

Computational Power of Restricted 2D Automata. My primary research focus for two-dimensional automata is an evaluation of the computational power of three-way and two-way two-dimensional automata in two ways: determining whether certain decision problems were decidable for certain models, and if a decision problem is decidable, analyzing the computational complexity of that problem.

In a paper on this topic [4], I found that the problems of determining whether one language was equivalent to another and determining whether one language was contained within another were both decidable for deterministic two-way two-dimensional automata. These results were obtained by applying a novel pumping-style argument to the dimensions of the input word to such an automaton. These were the first known positive decidability results for these decision problems in any two-dimensional automaton model. I further showed that the same problems were undecidable if we allowed the two-way two-dimensional automaton to be nondeterministic.

In addition, I considered the problem of determining whether a language is empty, or equivalently, whether a given automaton accepts no words. This problem is known to be decidable for both three-way and two-way two-dimensional automata. I refined these decidability results to show that, in the two-way case, the problem can be decided efficiently in deterministic polynomial time, while in the three-way unary case, the problem becomes NP-complete. This notable gap in complexity between these two models shows how, by making a change as small as reducing the number of directions of input head movement, certain decision problems can become tractable. My work in this area emphasizes that we contend with a crucial tradeoff between recognition power and desirable decidability/complexity properties when dealing with two-dimensional automata.



(a) Removing shared computations (b) Removing rows from word (c) Removing columns from word

Figure 1: Illustrations depicting various stages of the pumping-style argument from [4]

Investigating New Language Operations for 2D Automata. In addition to studying decision problems, we can also study closure properties: if we apply a given operation to two words from a given class, does the result still belong to the same class? If so, we say that the class is closed under the operation. The operation of concatenation joins two words together, and in one dimension, concatenation is straightforwardly defined. However, we encounter some challenges in two dimensions: namely, along which dimension do we concatenate words, and how do we match rows or columns of one word to another? Algebraically speaking, we also encounter a dilemma: unlike one-dimensional concatenation, two-dimensional concatenation does not induce a monoid structure over a set of words. Therefore, studying this particular operation gives us insights into the underlying structure of two-dimensional words and languages.

It was previously known that nondeterministic three-way two-dimensional automata are closed under row concatenation. In a recent paper [6], I investigated further closure properties of concatenation for three-way and two-way two-dimensional automata. My main contributions are twofold: first, I establish that both row and column concatenation operations are closed in the unary two-way case (notably, the first known positive closure result for column concatenation over any model) by using an argument wherein the moves of a pair of two-way one-dimensional automata are composed to simulate a two-dimensional automaton; and second, I show that three-way two-dimensional automata are not closed under a novel operation known as "diagonal concatenation", by simulating the computation of such an automaton using a classical finite automaton.

Connections between 2D Automata and 1D Languages. The recognition power of a given two-dimensional automaton model allows us to determine the kinds of words it is able to accept (in terms of either dimension or contents, or both). This information allows us to draw interesting connections between two-dimensional automata and one-dimensional language classes by applying an operation known as "projection" to the first row or first column of each word accepted by a given automaton.

Beginning in an earlier paper [4] and continuing in my most recent work [5], I classified the projections of languages accepted by two-dimensional automaton models in terms of familiar one-dimensional language classes (i.e., the Chomsky hierarchy). I found that the column projection of a unary three-way language can be recognized using only nondeterministic logspace: the first nontrivial space complexity upper bound for projection languages. I also showed that the projection of any two-way language is always regular, as is the row projection of a unary three-way language. In contrast to this result, I proved that four-way projection languages coincide exactly with the set of context-sensitive languages, establishing an interesting gap in recognition power while also allowing us to draw parallels to the more familiar linear-bounded automaton.

Another main contribution of my work [5] is the first descriptional complexity bounds for operations on two-dimensional languages. I obtained both upper and lower bounds on the number of states of a two-way two-dimensional automaton recognizing either the union of two languages or the diagonal concatenation of two languages. These results tie into my related work on language operations and closure properties, as I previously found these two operations to be the only operations known to be closed for this model.

Other Work. In addition to my work on two-dimensional formal language and automata theory, I have collaborated on research at the intersection of formal languages and bioinformatics. During my doctoral studies, I worked with researchers outside of my university on a formal language model of site-directed mutagenesis, a biological technique where polymerase chain reaction-based methods generate mutations in DNA. Our model, called site-directed insertion or SDI, is an overlapping insertion operation on words and languages that simulates site-directed mutagenesis by way of finding and matching prefixes/suffixes of one input word to substrings of another input word, then inserting the former word into the latter.

In our joint work [1], we investigated the SDI operation and established that deciding whether certain SDI-related properties apply to a pair of regular languages L_1 and L_2 can be done efficiently in polynomial time. Namely, we can efficiently decide the properties of "SDI-freeness", where no word from L_2 can be site-directed inserted into a word from L_1, and "SDI-independence", where site-directed insertion of any string into a word from L_1 does not produce a word from L_2.

Furthermore, we introduced the operations of maximal SDI and minimal SDI, where the maximal (resp., minimal) SDI of a word y into a word x is an insertion where the overlapping prefix/suffix of y into x cannot be extended (resp., reduced) in terms of length. We showed that applying maximal/minimal SDI to a pair of regular and finite languages preserves regularity. We also established that deciding the membership problem for maximal/minimal SDI can be done in polynomial time, and we gave nondeterministic state complexity bounds on the size of automata recognizing maximal/minimal SDI languages. Altogether, our contributions in this paper show that certain biological operations can be modelled within formal language theory using classical finite automata, and questions relating to these operations can be solved efficiently.

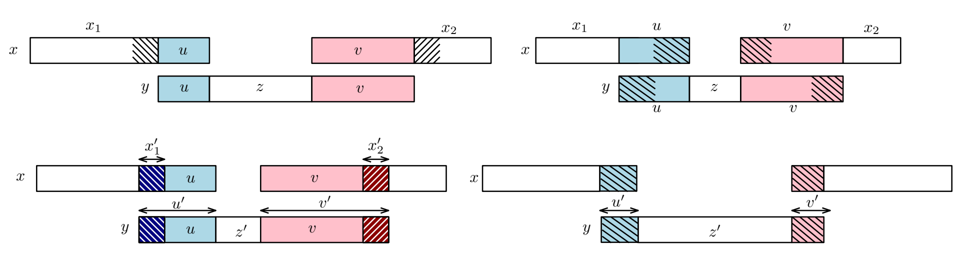


Figure 2: Illustrations of SDI (top-left), max-SDI (bottom-left), and min-SDI (right) operations from [1]

Future Directions. My long-term research plan can be divided into two branches: first, I intend to continue investigating the theoretical properties of two-dimensional automaton models and variants thereof, establishing further descriptional and computational complexity bounds in an effort to characterize fully the power of these models; and second, I will study applications of two-dimensional automata to other fields of computer science and beyond and to establish whether this or similar computational models can give a tangible advantage in computationally-intensive tasks. The first branch can be pursued in the short-term (1–3 years), while the second branch is longer-term (3–5 years) as any potential applications will rely on the theoretical properties established in my earlier research.

In each of my papers, I have identified a selection of problems spurring from my contributions that remain open. Most of these problems relate to closure properties and decision problems, as discussed earlier: for instance, it is presently unknown whether some problems are decidable for certain three-way two-dimensional automaton models. Of the problems that are decidable, we do not presently know whether efficient algorithms exist to decide each of these problems, or even to which computational complexity class these decision problems belong. One particular avenue for further research is the study of descriptional complexity I initiated in my recent paper [5]. A number of problems on state complexity of two-dimensional automata await further study, which I anticipate will guide a portion of my future work.

Research in two-dimensional automata (and formal languages) has immediate applications to applied areas of computer science such as pattern recognition, pattern matching, image processing, coding theory, and discrete event systems. Indeed, some of my past work in formal language theory has investigated practical applications such as efficiently identifying repeating patterns and segments within a block of two-dimensional data [2], and I intend to continue pursuing algorithmic applications of my work as I build up the theoretical underpinnings. However, results in this vein of research can also be applied to areas that are perhaps not immediately associated with theoretical computer science. For example, certain families of two-dimensional languages can be used to model the behaviour of symbolic dynamic systems in physics. Although I have not yet investigated applications of my research beyond computing, such connections to other areas of science invite the opportunity for fruitful collaborations with researchers in other departments.

References

[1] Da-Jung Cho, Yo-Sub Han, Kai Salomaa, and **Taylor J. Smith**. Site-directed insertion: Language equations and decision problems. *Theoret. Comput. Sci.*, 798:40–51, 2019.

[2] Guilhem Gamard, Gwenaël Richomme, Jeffrey Shallit, and **Taylor J. Smith**. Periodicity in rectangular arrays. *Inform. Process. Lett.*, 118:58–63, 2017.

[3] **Taylor J. Smith**. Two-dimensional automata. Technical report 2019-637, Queen's University, Kingston, 2019.

[4] **Taylor J. Smith** and Kai Salomaa. Decision problems for restricted variants of two-dimensional automata. In *Proc. of CIAA 2019*, volume 11601 of *LNCS*, pages 222–234. Springer-Verlag, 2019.

[5] **Taylor J. Smith** and Kai Salomaa. Recognition and complexity results for projection languages of two-dimensional automata. In *Proc. of DCFS 2020*, volume 12442 of *LNCS*, pages 206–218. Springer-Verlag, 2020.

[6] **Taylor J. Smith** and Kai Salomaa. Concatenation operations and restricted variants of two-dimensional automata. In *Proc. of SOFSEM 2021*, volume 12607 of *LNCS*, pages 147–158. Springer-Verlag, 2021.