A research proposal for the computational complexity of minimizing reducible sofic shifts

Justin Cai

April 23, 2020

Abstract

In symbolic dynamics, a certain class of objects called sofic shifts are sets of bi-infinite sequences that come from labeled graphs, such that each letter of the sequence is the label of an edge in a bi-infinite walk around the graph. If one wanted to use sofic shifts to model something, it would be desirable to have a graph that is as small as it can be while still presenting the same sequences. While the problem of how hard it is to compute this minimal graph for shifts with a certain property called irreducibility is known, the hardness of computing the same problem for shifts that do not have this property is unknown.

1 Background

A *full shift* is the set of all bi-infinite sequences over a finite alphabet $\mathcal{A}$. A *graph* G is a finite set of *vertices* $\mathcal{V} = \mathcal{V}(G)$ and a finite set of edges $\mathcal{E} = \mathcal{E}(G)$ with each edge e starting at a vertex $i(e) \in \mathcal{V}$ and terminating at a vertex $t(e) \in \mathcal{V}$. A bi-infinite walk on G is a bi-infinite sequence of edges such that the terminating vertex of each edge is the inital vertex of the next edge. The set of all bi-infinite walks on G is called the *edge shift* X_G. A *labeled graph* $\mathcal{G}$ is a graph G equipped with a *labeling* $\mathcal{L} : \mathcal{E}(G) \to \mathcal{A}$, which assigns each edge e from G a label $\mathcal{L}(e)$ from a finite alphabet $\mathcal{A}$. If x is a bi-infinite walk on G, then the *label of the bi-infinite walk* $\mathcal{L}_\infty(x)$ is the bi-infinite sequence of the labels of x. The set of all labels of bi-infinite walks is called *presentation of* $\mathcal{G}$, denoted $\mathsf{X}_\mathcal{G}$. A subset X of a full shift is a *sofic shift* if $X = \mathsf{X}_\mathcal{G}$ for some labeled graph $\mathcal{G}$. A labeled graph $\mathcal{G}$ is a *presentation* of X or *presents* X if $X = \mathsf{X}_\mathcal{G}$.

For example, let X be the set of bi-infinite sequences over $\{0, 1\}$ such that there is an even number of 0's between any two 1's. Then $X = \mathsf{X}_\mathcal{G}$, where $\mathcal{G}$ is any labeled graph in Figure 1. This shift is known as the *even shift*.

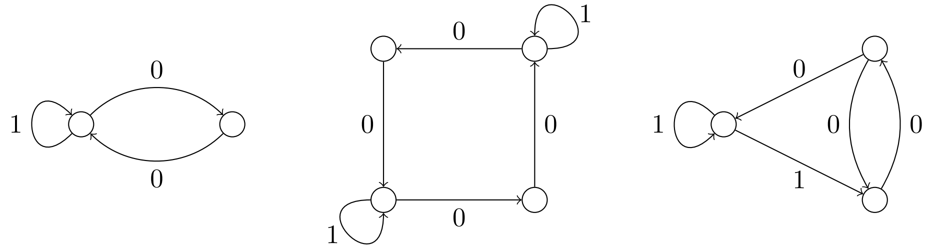

Figure 1: Presentations of the even shift

A *block* is a finite sequence of symbols over an alphabet. Let x be a bi-infinite from a sofic shift. We say the a block w *occurs in* x if there exists integers i, j such that $x_i x_{i+1} \dots x_j = w$. The *language* of a sofic shift $\mathcal{B}(X)$ is the collection of blocks that occur in any bi-infinite sequence in X. A sofic shift is *irreducible* if for any pair of blocks $u, v \in \mathcal{B}(X)$, there exists another block $w \in \mathcal{B}(X)$ such that $uwv \in \mathcal{B}(X)$.

A presentation is *right-resolving* if for each vertex in the presentation, the labels of the outgoing edges of that vertex are all distinct. For example, the left and middle graphs in Figure 1 are right-resolving while the right graph is not right-resolving. A presentation is *irreducible* if for each pair of vertices in the graph, there exists a path in the graph from the first vertex to the second vertex and path from the second vertex to the first vertex. A presentation that is not irreducible is *reducible*. Each presentation in Figure 1 is irreducible, while the presentation in Figure 2 is reducible.

For a presentation $\mathcal{G}$, the *follower set* of a vertex I is the labels of all (finite) paths that originiate from that vertex, denoted $F_{\mathcal{G}}(I)$. The presentation is *follower-separated* if the follower sets of $\mathcal{G}$ are distinct; that is, for any pair of vertices $I, J \in \mathcal{V}$ such that $I \neq J$, then $F_{\mathcal{G}}(I) \neq F_{\mathcal{G}}(J)$

A *minimal* presentation of a sofic shift X is a presentation such that it has the fewest vertices among all presentations of X; that is, a presentation $\mathcal{G}$ is minimal for X such that for all presentations $\mathcal{G}'$ of X, $|\mathcal{V}(\mathcal{G})| \leq |\mathcal{V}(\mathcal{G}')|$.

2 Minimization of reducible presentations

From [Lin+95] corollary (3.3.20), given an irreducible right-resolving presentation, we can find the minimal right-resolving presentation by merging vertices in the presentation that have the same follower set, creating a follower-seperated presentation. However, this does not work for reducible presentations, as being follower-separated does not imply minimality. We can see this with this presentation of the even shift:

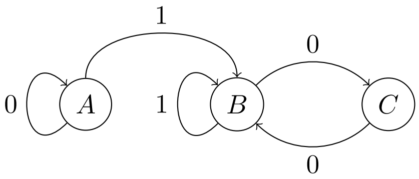

Figure 2: A reducible, follower-separated presentation of the even shift

The block 01 does not appear in the follower set of B, but does for the follower set of A, so $F_{\mathcal{G}}(A) \neq F_{\mathcal{G}}(B)$. The follower set of C is distinct from the follower sets of A and B, as any word that starts with 1 that appears in $F_{\mathcal{G}}(A)$ and $F_{\mathcal{G}}(B)$ does not appear in $F_{\mathcal{G}}(C)$, so $F_{\mathcal{G}}(A) \neq F_{\mathcal{G}}(C)$ and $F_{\mathcal{G}}(B) \neq F_{\mathcal{G}}(C)$. Hence, the presentation is follower-separated. This graph also presents the even shift as any walk that visits A has a corresponding walk that only visits B and C (and the BC subgraph presents the even shift). Since this presentation is follower-separated, it would be minimized if (3.3.20) applied to reducible presentations, but since the graph presents the even shift, it is not minimal (as a minimal presentation of the even shift has 2 vertices).

Therefore, an algorithm for minimizing reducible presentations is still to be desired. However, a polynomial time algorithm might be unlikely, as this research conjectures that the problem of deciding if such presentation can be minimized is as hard as the graph isomorphism problem.

The GI complexity class is the set of decision problems that have a polynomial-time Turing reduction to the graph isomorphism problem (a *graph isomorphism* is an invertible mapping from the vertices of one graph to the vertices of another such that two vertices are connected if and only if they are connected under the isomorphism; the graph isomorphism problem asks given two graphs, decide whether there exists an isomorphism between the graphs). Currently, it is unknown if GI is contained in P nor is NP-complete [KST12].

Define the *minimal reducible presentation* problem, or minRP, as given a reducible presentation, decide if it is a minimal presentation. The complement problem co-minRP asks, given a reducible presentation, to decide if it is not minimal. A problem is GI-*hard* if there exists a polynomial-time Turing reduction from any problem in GI to that problem. Now, this research proposes the following conjecture:

Conjecture. *The complement of the minimal reducible presentation problem,* co-minRP*, is* GI-*hard.*

A reduction candidate I've looked at is to take two irreducible, minimal, right-resolving presentations that are isomorphic, and then take an edge from one graph and change its terminal vertex to a vertex in the other graph, creating a reducible graph. For example, if we take two minimal presentations of the even shift, we can connect them like so:

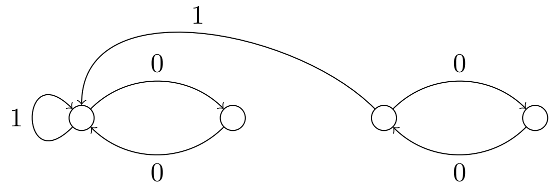

Figure 3: Connecting two even shifts to create a reducible, non-minimal even shift

The resulting graph still presents the even shift, so the presentation is not minimal. Hence one could propose that two graphs are isomorphic if and only if there exists an edge in one graph and a vertex in the other such that if you connect them, then the resulting graph is not minimal. For the reduction to hold, we'd still need to show that if two graphs are not isomorphic, then for any edge in one graph and any vertex in the other, the resulting graph from the connection is always minimal. However, there exists a counterexample:

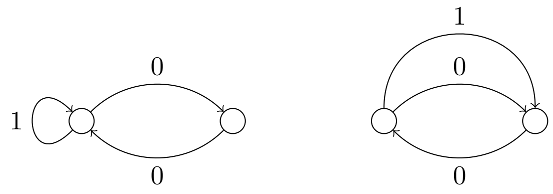

Figure 4: Two non-isomorphic graphs that can be connected like Figure 3

You can take the edge labeled 1 on the right graph and connect it the same way as in Figure 3. Thus, we have two graphs that are not isomorphic but there exists an edge and vertex such that the connected graph is not minimal.

Instead of modifying one of the graphs, one could augment the graphs by adding an edge with a label distinct to the labels of the two graphs from one of the graphs to the other and then adding a self loop with the same label on the vertex that was connected to. For example, starting again with two minimal even shifts,

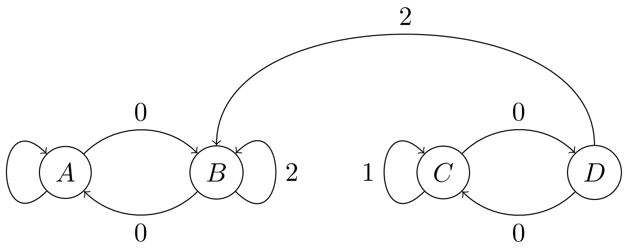

Figure 5: A proposed reduction

The shift this graph presents is the same as the presentation of the subgraph consisting of vertex A and B and the edges who begin in those vertices, as any walk that visits C and D has a corresponding walk from the AB subgraph.

References

[Lin+95] Douglas Lind et al. *An introduction to symbolic dynamics and coding*. Cambridge university press, 1995.

[KST12] Johannes Kobler, Uwe Schöning, and Jacobo Torán. *The graph isomorphism problem: its structural complexity*. Springer Science & Business Media, 2012.