Vol.2014-MPS-98 No.27
Vol.2014-BIO-38 No.27
2014/6/27

Grammar-based Compression for Multiple Trees Using Integer Programming

Yang Zhao[1,a)] Morihiro Hayashida[1,b)] Yue Cao[1,c)] Jaewook Hwang[1,d)] Tatsuya Akutsu[1,e)]

Abstract: Many tree structures can be found in nature and organisms. It is considered that such trees are constructed by some rules. In our previous study, bisection-type grammar-based compression methods for ordered and unordered single trees have been developed. Here, these methods find construction rules for one tree. On the other hand, specified construction rules can be contributed to generate several similar trees.
In this technical report, hence, we develop a method to find common rules generating multiple kinds of trees based on the previous method using integer programming. We apply our proposed method to several glycans that are one of important molecules in celluar systems and are regarded as tree structures. As a result, our method successfully found the minimum grammar and several common rules among these glycans.

1. Introduction

Many tree structures can be found in nature and organisms. We focus on finding construction rules for such tree structures. For that purpose, we use *simple elementary unordered tree grammar* (SEUTG) [1], and reduce our problem to the problem of finding the minimum SEUTG for tree structures. In our previous study, we proposed methods to find the minimum tree grammar using integer programming (IP) [2]. Here, the proposed methods compress single trees. On the other hand, specified construction rules can be contributed to generate several similar trees. In this study, we try to find common construction rules among several similar trees by extending the previous method. Then, we apply our method to several glycans that are attached to proteins and express essential functions in cellular system. As a result, our method successfully found the minimum SEUTG and several common rules among these glycans.

2. Method

In this section, we briefly review the tree grammar SEUTG and explain the extended IP formulation for multiple trees.

2.1 Simple Elementary Unordered Tree Grammar (SEUTG)

Simple elementary unordered tree grammar (SEUTG) is a bisection-type grammar as shown in Fig. 1, where trees are unordered, rooted, and edge-labeled, capital letters indicate non-terminal symbols, and small letters indicate terminal symbols.

[1] Bioinformatics Center, Institute for Chemical Research, Kyoto University, Gokasho, Uji, Kyoto 611–0011, Japan
[a)] tyoyo@kuicr.kyoto-u.ac.jp
[b)] morihiro@kuicr.kyoto-u.ac.jp
[c)] caoyue@kuicr.kyoto-u.ac.jp
[d)] hwangjw@kuicr.kyoto-u.ac.jp
[e)] takutsu@kuicr.kyoto-u.ac.jp

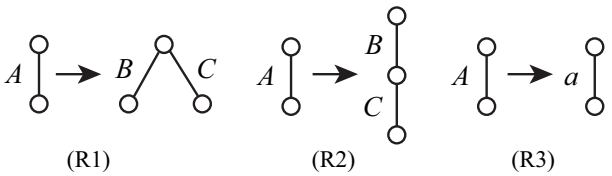

Fig. 1 Rules without tags in simple elementary unordered tree grammar (SEUTG). (R1) horizontal bisection rule. (R2) vertical bisection rule. (R3) replacement rule with a terminal symbol. Capital letters, 'A', 'B', and 'C' indicate non-terminal symbols, and small letter, 'a', indicates a terminal symbol.

(R1) shows a horizontal bisection rule, (R2) shows a vertical bisection rule, and (R3) shows a replacement rule from a non-terminal symbol to a terminal symbol.

2.2 Extension to Multiple Trees

In the previous method, the IP finds a grammar with fixed number of non-terminal symbols. Also in the extended method, we fix the number of non-terminal symbols denoted by m, and examine $m = 1$ to the minimum number to solve the problem.

Let N and T_α be the number of given trees and α-th tree, respectively. Then, by extending the previous one, we have the IP formulation as shown in Fig. 2. Here, variable $x_{\alpha,1,\epsilon,ch(1)}$ appearing in the objective function becomes 1 if α-th tree is constructed by an SEUTG, otherwise 0. The objective function that is the sum of these variables becomes N if and only if all N trees are constructed using m non-terminal symbols.

$T_{\alpha,i,t,C}$ denotes the subtree rooted at i in α-th tree, where C denotes the subset of children of vertex i in the subtree, and vertex t is labeled as a tag if t is not represented as ϵ. A tag means that the vertex with the tag is attached to the root of another tree. The root of each tree is numbered as 1. $ch(i)$ denotes the set of all children of i in the original tree. Thus, the variable $x_{\alpha,1,\epsilon,ch(1)}$ in the objective function corresponds to the α-th tree $T_{\alpha,1,\epsilon,ch(1)}$.

Eq. (1) and (2) in Fig. 2 correspond to (R3). Each edge in

$$\text{Maximize} \quad \sum_{\alpha=1}^{N} x_{\alpha,1,\epsilon,ch(1)}$$

Subject to

$$x_{\alpha,i,\epsilon,\{j\}} = 1 \qquad \text{for all } i,\ j \in ch(i)\ (|ch(j)| = 0) \tag{1}$$

$$x_{\alpha,i,j,\{j\}} = 1 \qquad \text{for all } i,\ j \in ch(i)\ (|ch(j)| > 0) \tag{2}$$

$$x_{\alpha,i,\epsilon,C} \le \sum_{C'(\neq\emptyset)\subset C} y^{ho}_{\alpha,i,\epsilon,C',C-C'} + \sum_{t\in I(T_{\alpha,i,\epsilon,C})} y^{ve}_{\alpha,i,\epsilon,C,t} \qquad \text{for all } i,\ C \subseteq ch(i) \tag{3}$$

$$y^{ho}_{\alpha,i,\epsilon,C',C-C'} \le \frac{1}{2}(x_{\alpha,i,\epsilon,C'} + x_{\alpha,i,\epsilon,C-C'}) \tag{4}$$

$$y^{ve}_{\alpha,i,\epsilon,C,t} \le \frac{1}{2}(x_{\alpha,i,t,C} + x_{\alpha,t,\epsilon,ch(t)}) \tag{5}$$

$$x_{\alpha,i,j,C} \le \sum_{C'(\neq\emptyset)\subset C} y^{ho}_{\alpha,i,j,C',C-C'} + \sum_{t\in an(j)-\{i\}} y^{ve}_{\alpha,i,j,C,t} \qquad \text{for all } i,\ j \in I(T_{\alpha,i,\epsilon,C}),\ C \subseteq ch(i) \tag{6}$$

$$y^{ho}_{\alpha,i,j,C',C-C'} \le \frac{1}{2}(x_{\alpha,i,\epsilon,C'} + x_{\alpha,i,j,C-C'}) \tag{7}$$

$$y^{ve}_{\alpha,i,j,C,t} \le \frac{1}{2}(x_{\alpha,i,t,C} + x_{\alpha,t,j,ch(t)}) \tag{8}$$

$$z_u \ge \frac{1}{n} \sum_{\{\alpha,i,j,C:es(T_{\alpha,i,j,C})=u\}} x_{\alpha,i,j,C} \qquad \text{for all distinct Euler strings } u \text{ of } T_{\alpha,i,j,C} \tag{9}$$

$$\sum_{u} z_u = m \tag{10}$$

Fig. 2 IP formulation for finding generation rules of multiple unordered trees in SEUTG.

Table 1 Result on the minimum number of non-terminal symbols for glycans, G02677, G03661, and G03664.

Glycan	#vertices	min # non-terminal symbols
G02677	15	13
G03661	15	17
G03664	17	16
G02677, G03661, G03664	47	31

N trees are constructed by (R3). Eq. (3) and (6) represent that a subtree is constructed by (R1) or (R2). $y^{ho} = 1$ if a horizontal bisection can be used for construction of the subtree, otherwise 0. $y^{ve} = 1$ if a vertical bisection can be used. $I(T)$ denotes a set of internal vertices (neither a root or leaves) in tree T. $an(j)$ denotes a set of ancestors of j ($j \notin an(j)$, and suppose $an(\epsilon) = \emptyset$). Eq. (4), (5), (7), and (8) represent that both of two subtrees that become sources of bisection rules are constructed. Eq. (9) represents that all the same subtrees are represented by a non-terminal symbol. $es(T)$ denotes the Euler string of the unordered tree T, where for a tagged tree, the tagged edge with label A is transformed into $A\tau\overline{A}$ using a special symbol τ representing the tag.

On the other hand, there is another approach for finding the minimum SEUTG for N trees. It adds a special root vertex, attaches N trees to the root, and applies the previous IP formulation. This approach, however, increases the number of variables in the IP, and may enlarge the execution time.

3. Computational Experiment

For evaluation of our method, we used glycans that are regarded as rooted tree structures. It is known that glycans are attached to proteins and express essential functions in cellular systems. We used three glycans, G02677, G03661, and G03664 in KEGG Glycan database [3], where each edge is labeled with the sugar of the endpoint far from the root. We applied our method to each glycan and these three glycans.

Table 1 shows the result on the minimum number of non-terminal symbols m in the IP. The minimum number for these three glycans was smaller than the sum of the minimum number for each glycan, that is, 13+17+16=46. It suggests that our method successfully found several common rules among the glycans. Fig. 3 shows examples of extracted patterns commonly appearing in two or three glycans of G02677, G03661, G03664. It should be noted that the trees in this figure are equivalent except the difference of a tag.

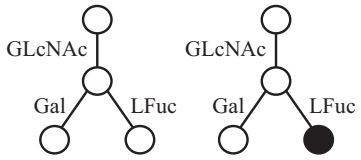

Fig. 3 Example of extracted patterns appearing in two or three glycans of G02677, G03661, G03664.

4. Conclusion

We proposed the method to find the minimum simple elementary unordered tree grammar (SEUTG) for multiple trees. Since glycans have important functions in celluar systems and are regarded as rooted, unordered trees, we applied our method to several glycans. As a result, our method found the minimum SEUTG and several common rules among the glycans. The minimum number of non-terminal symbols for these three glycans was smaller than the sum of the minimum number for each glycan. As future work, we would like to apply our method to more glycans and other tree structures.

Acknowledgments

This work was partially supported by Grants-in-Aid #26240034, #24500361, and #25-2920 from MEXT, Japan.

References

[1] Akutsu, T.: A bisection algorithm for grammar-based compression of ordered trees, *Information Processing Letters*, Vol. 110, pp. 815–820 (2010).

[2] Zhao, Y., Hayashida, M. and Akutsu, T.: Integer programming-based method for grammar-based tree compression and its application to pattern extraction of glycan tree structures, *BMC Bioinformatics*, Vol. 11, No. Suppl 11, p. S4 (2010).

[3] Hashimoto, K., Goto, S., Kawano, S., Aoki-Kinoshita, K., Ueda, N., Hamajima, M., Kawasaki, T. and Kanehisa, M.: KEGG as a glycome informatics resource, *Glycobiology*, Vol. 16, No. 5, pp. 63R–70R (2006).