The Decomposition Algorithm of Skew-symmetrizable Exchange Matrices

Weiwen Gu

Abstract

Some skew-symmetrizable integer exchange matrices are associated to ideal (tagged) triangulations of marked bordered surfaces. These exchange matrices admits unfoldings to skew-symmetric diagrams. We develop an combinatorial algorithm that determines if a given skew-symmetrizable matrix is of such type. This algorithm generalizes the one in [1]. As a corollary, we use this algorithm to determine if a given skew-symmetrizable matrix has finite mutation type.

1 Introduction

With ~~To~~ some triangulations have of surfaces invariant under finite group symmetries, we associate quivers ~~graphs~~ whose edges ~~admit~~ positive integer weights. Such graphs are also associated with matrices with integer entries. We call such graphs or their associated matrices *s-decomposable* (see ~~exact~~ definition 5.) ~~Recall that a skew-symmetric integer matrix with all entries equal to $1, -1, 2, -2$ or 0 is associated to a graph called *quiver*.~~ Quiver *mutation* is ~~an operation on quiver~~ defined in [2]. The collection of mutation-equivalent quivers to a given quiver G is called the mutation class of G. We say a quiver is *mutation finite* or has *finite mutation type* if its mutation class is finite.

A quiver or its associated skew-symmetric matrix is said to be block-decomposable if the quiver can be obtained by combining pieces of graphs isomorphic to six types of quivers, called elementary blocks, by a particular way of gluing (see exact definition 1). It is proved in [3] that a quiver has finite mutation type if and only if it is either block-decomposable or is of one of the 11 exceptional types. It is also proved ~~in the same article~~ that a quiver is block-decomposable if and only if it is the associated adjacency graph of an ideal triangulations of a bordered surface with marked points.

via [Fomin Shapiro Thurston]

An $n \times n$ integer matrix B is said to be *skew-symmetrizable* if there exists an $n \times n$ integer diagonal matrix D such that BD is skew-symmetric. Mutation and nutation class are also defined ~~on~~ for skew-symetrizable matrices. Our goal is to establish an combinatorial algorithm that determines if a given oriented graph whose edges ~~are oriented and~~ are equipped with integer weights is s-decomposable, ~~and~~ thus, provid~~es~~ing a tool to find if its associated skew-symmetrizable matrix has finite mutation type. Each skew-symmetrizable exchange matrix is associated to a ~~one dimensional complex~~ diagram ~~whose edges are oriented and admit positive integer weights.~~ with oriented edges equipped with positive integer weights

The notion of 7 additional blocks

s-decomposability is a generalization of *block-decomposability* (see Definition 5 and Table 2). A skew-symmetrizable exchange matrix is said to be s-decomposable if it can be obtained by gluing both elementary blocks and ~~new~~ blocks by the rules in Def. 1 and Def. 5.

In [1], we provided an algorithm linear in the size of quiver G that determines if G is block-decomposable. As a corollary, we obtained ~~proved that~~ for any skew-symmetric integer matrix B, ~~there exists~~ an algorithm linear in the size of B ~~that~~ determining ~~determines~~ if B has finite mutation type. The algorithm we describe ~~establish~~ in this article is a generalization of the one in [1]. This paper ~~The generalization~~ is inspired by [2], in which the authors generalize the result of ~~in~~ [3] to the ~~adjacency graphs associated to~~ skew-symmetrizable *exchange matrices*. (diagrams). The following results are proved in [2]:

1. There is a one-to-one correspondence between s-decomposable skew-symmetrizable graphs with fixed block decomposition and ideal tagged triangulations of marked bordered surfaces with fixed tuple of conjugate pairs of edges.

2. A skew-symmetrizable $n \times n$ matrix, $n \geq 3$, that is not skew-symmetric, has finite mutation class if and only if ~~its associated graph~~ diagram is either s-decomposable or mutation-equivalent to one of seven types.

3. Any s-decomposable diagram admits an *unfolding* to a diagram associated to ideal tagged triangulation of a marked bordered surface.Any mutation-finite matrix with non-decomposable diagram admits an unfolding to a mutation-finite skew-symmetric matrix.

According to the above theorems, if G is the diagram associated to a skew-symmetrizable exchange matrix M, and T is the ideal tagged triangulation corresponding to a particular s-decomposition G_{dec} of G, then an unfolding to G_{dec} defines a skew-symmetric diagram obtained by gluing of unfoldings of corresponding blocks. We design an ~~The generalized~~ algorithm that determines if a given graph is s-decomposable, and for each possible decomposition, finds the associated ideal tagged triangulation of bordered surface with marked points. ~~Therefore, our algorithm determines if a given skew-symmetrizable matrix has finite mutation type.~~ Moreover, this algorithm is linear in the size of the given matrix. In order to determine if a given skew-symmetrizable matrix has finite mutation type it remains to check if it is of one of the 11 (for skew-symmetric) or 7 (for skew-symmetrizable) types. Since this requires a bounded number of operations we obtain a linear algorithm

2 Definitions

In this section, we introduce definitions ~~required to establish our algorithm, and give~~ a brief discription ~~procedure~~ of the algorithm. For convenience, we denote an edge that connects nodes x, y by $\overline{xy}$ if the orientation of this edge is unknown or irrelevant, $\overrightarrow{xy}$ if the edge is directed from x to y, and $\overleftarrow{xy}$ ~~if from y to x~~ otherwise.

Definition 1. We recall that a diagram (or graph) is *block-decomposable* (or *decomposable*) if it is obtained by gluing elementary blocks of ~~in~~ Table 1 by the following *gluing rules*:

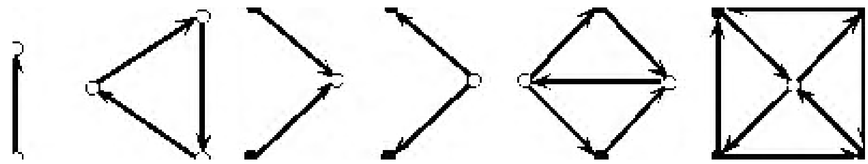
1. Two white nodes of two different blocks can be identified. As a result, the graph becomes a union of two parts; the ~~The~~ common node is colored ~~becomes~~ black. A white node can neither be identified to itself nor with another node of the same block.

If two white nodes x, y of one block (endpoints of edge $\vec{xy}$) are identified with two white nodes p, q of another block (endpoints of edge $\vec{pq}$) x with p, y with q correspondingly then a multiedge of weight 2 is formed, and nodes $x=p$, $y=q$ are black.

2. A black node can not be identified with any other node.

3. If an edge $a = x \to y$ with two white nodes (x, y) is glued to another edge $b = p \to q$ with two white nodes (p, q) in the following way: x is glued to p and y is glued to q, then a multi-edge is formed, and the nodes $x = p$, $y = q$ become black.

4. If an edge $a = x \to y$ with two white nodes x, y is glued to another edge $b = q \to p$, then both edges are removed after gluing, the nodes $x = p$, $y = q$ become black. We say that edges annihilate each other. please change accordingly.



Spike	Triangle	Infork	Outfork	Diamond	Square

Table 1: Elementary Blocks

you need to write here s.th about cluster algebras

why signed adjacency?

Definition 2. Let $B(G) = (b_{ij})$ be the skew-symmetric matrix whose rows and columns are labeled by the vertices of G, and whose entry b_{ij} is equal to the number of edges going from i to j minus the number of edges going from j to i. We say $B(G)$ is the *signed adjacency matrix associated to G*, ~~If~~ if an matrix $B = B(G)$, we say G is the *oriented adjacency graph associated to B.*

Remark 1. By definition, the associated matrix to a ~~decomposable~~ (any?) graph is skew-symmetric.

Definition 3. A *seed* is a pair (f, B), where $f = f_1, \ldots, f_n$ form a collection of algebraically independent rational functions of n variables $x_1, \ldots, x_n$, and B is a skew-symmetrizable matrix. The part f of seed (f, B) is called *cluster*, elements f_i are called *cluster variables*, and B is called *exchange matrix.*

add here defn of quiver/matrix mutation

It is proved in [3] that an skew-symmetric exchange matrix has finite mutation type if it is associated to a decomposable graph or one of the 11 exceptional types. To generalize the results to skew-skymmetrizable matrix, we need the following definitions:

Definition 4. A *diagram* (or *graph*) S associated to a skew-symmetrizable integer matrix B is an oriented graph with weighted edges obtained in the following way: Suppose $B = (b_{ij})_{i,j=1}^{n}$. Vertices of S are labeled by $[1, \ldots, n]$. If $b_{ij} > 0$, we join vertices i and j by an edge directed from i to j and assign to this edge weight $-b_{ij}b_{ji}$.

Definition 5. If a graph G can be obtained by gluing both elementary blocks and new blocks in Table 2 by the gluing rules in Definition 1 and the following new rules, we say the graph is *s-decomposable*:

!

Discuss possibility to change weight to $\sqrt{-b_{ij}b_{ji}}$

1. If the graph has multiple edges containing n parallel edges, replace the multiple edge by an edge of weight $2n$. For example, if we glue two parallel spikes of the same direction, we get an edge of weight 4 (see Figure 1).

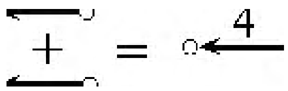


Figure 1: Edge of Weight 4

2. All single edges have weight 1.

Remark 2. If G is obtained by gluing blocks by the above rules, G is associated to a ideal tagged triangulation of bordered surface with marked points obtained by the gluing of the pieces of surfaces associated to the blocks, two arcs are glued together iff the corresponding nodes are glued together in the associated blocks. references here.

Remark 3. According to the above rules, the weight of any edge in a decomposable graph can only be $1, 2$ or 4. All edges of weight 2 can only be obtained from blocks in Table 2. Moreover, since all edges of weight 2 contains at least one black endpoint, we can never obtain an edge of weight 4 from ~~two~~ edges of weights 2. Moreover, an edge of weight 4 can only be obtained from **IV**or by Figure 1.

The *unfolding* procedure is defined as follows (see section 4 in [2]): Suppose that we have a chosen disjoint index sets: $E_1, E_2, \ldots, E_n$, with $|E_i| = d_i$. Denote $m = \sum_{i=1}^{n} d_i$. To each matrix B' mutation-equivalent to a given skew-symmetrizable $m \times m$ matrix B, a skew-symmetric matrix $\widehat{B}'$ indexed by $\bigcup_{i=1}^{n} E_i$ is defined so that the following conditions are satisfied:

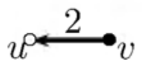
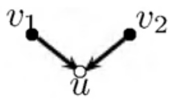
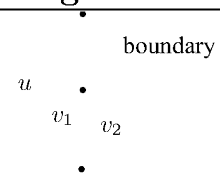
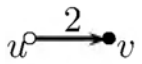
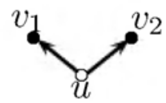
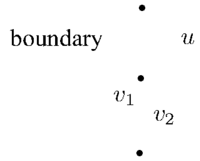
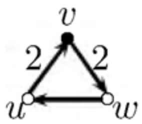
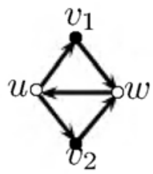
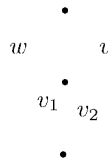
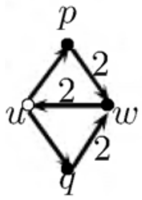
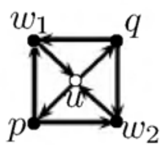
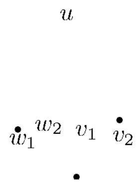
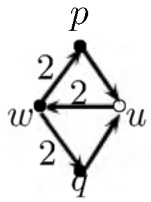
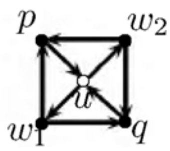
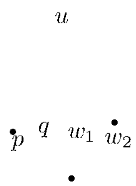
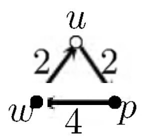
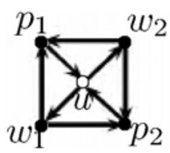
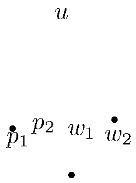
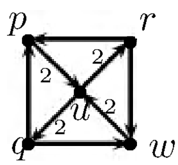
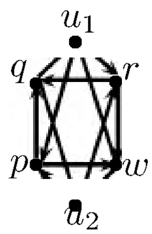
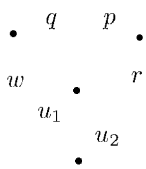
1. the sum of entries in each column of each $E_i \times E_j$ block of $\widehat{B}'$ equals to b_{ij};

2. if $b_{ij} \geq 0$, then the $E_i \times E_j$ block of $\widehat{B}'$ has all entries non-negative.

Define a composite mutation $\widehat{\mu}_i = \prod_{j \in E_i} \mu_j$ on $\widehat{B}'$. If C is the skew-symmetric matrix constructed from B satisfying the above conditions, we say C is an unfolding if for any B' mutation-equivalent to B, $\widehat{\mu}_i(\widehat{B}') = \widehat{\mu(B')}$.

A geometric interpretation of mutations on the blocks in Table 2 is given in Lusztig's [4]. Abusing the notation, we say the new blocks are the *foldings* of their corresponding unfoldings. Each unfolding represents an ideal tagged triangulation of bordered surface with marked points (see pictures in [2], (Table 7.1)). Each of these unfoldings except the last one corresponds to triangulations with two edges inside a digon (or monogon), one of the edges is tagged and the other is untagged. This pair of edges are said to be *conjugate*. Conjugate

Table 2: Blocks of Unfolding

	New Blocks	Unfolding	Triangulation
Ia:			
Ib:			
II:			
IIIa:			
IIIb:			
IV:			
V:			

edges represent the same vertex in ~~corresponding~~ the foldings. Mutation~~s~~ ~~on~~ of the ~~folding~~ folded vertex corresponds to the flips of both edges in the conjugate pair. *Composite flip* of ~~a~~ the triangulation corresponding to an unfolding diagram is defined as a collection of flips in all edges that represent~~s~~ all vertices ~~whose folding is the same vertex~~ in the set E_i. Note that any two flips in a composite flip commute, see Figure 2.

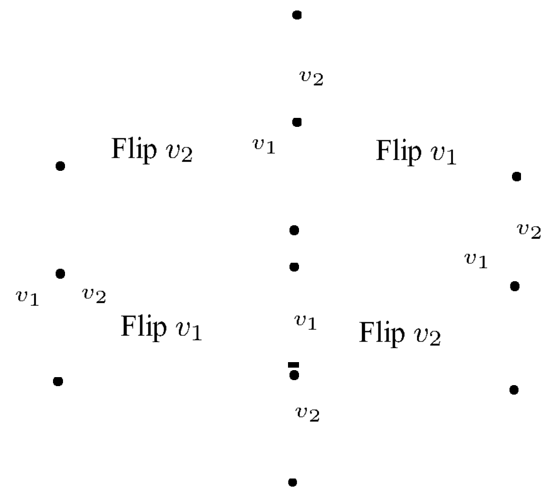


Figure 2: Composite Flip

In [2], the following theorem is proved:

Theorem 1. *Any s-decomposable diagram admits an unfolding to a diagram arising from ideal tagged triangulation of a marked bordered surface. Any mutation-finite matrix with non-decomposable diagram admits an unfolding to a mutation-finite skew-symmetric matrix.*

Given an s-decomposable diagram with a fixed decomposition, there is a unique tagged triangulation of a marked bordered surfaces with chosen tuples of conjugate pairs of edges. This surface can be obtained by gluing ~~of arcs in the triangulations~~ pieces of surfaces represent~~ed~~ing ~~by~~ unfoldings of corresponding blocks in the decomposition along edges corresponding to glued white vertices. The construction is invariant under mutation: ~~M~~mutating the diagram means performing composite flips to the original triangulations. Furthermore, the following theorems are proved in [2]:

Theorem 2. *There is a one-to-one correspondence between s-decomposable skew-symmetrizable diagrams with fixed block decomposition and ideal tagged triangulations of marked bordered surfaces with fixed tuple of conjugate pairs of edges.*

Theorem 3. *A skew-symmetrizable $n \times n$ matrix, $n \geq 3$, that is not skew-symmetric, has finite mutation class if and only if its diagram is either s-decomposable or mutation-equivalent to one of seven types.*

(non skew-symmetric)

that

By the previous theorems~~, if we need~~ to check ~~if~~ a given skew-symmetrizable matrix has finite mutation type, we can first check if it is mutation-equivalent to one of the seven types of diagrams. If not, we can further check if the associated adjacency graph is s-decomposable. We develop an algorithm ~~next to~~ below that determines if ~~any graph~~ a given diagram is s-decomposable.

For convenience, we need the following definition:

number these rules and refer to numbers

Definition 6. Suppose N is a subgraph of G with all its nodes colored white or black. If there exists another quiver M with all its nodes colored white or black, such that G can be obtained by gluing M to N by the rules in definitions 1 and 5, we say N is ~~a~~ *colored subgraph* of G. A *neighborhood* of o is a colored subgraph of G that contains node o. We say a colored subgraph N of G is *decomposable* if there exists an s-decomposable or decomposable graph $\widetilde{G}$ that contains N as a colored subgraph. A colored subgraph N of G is said to be *indecomposable* if any graph that contains N as a colored subgraph is ~~not~~ neither s-decomposable nor decomposable. We say a colored subgraph N is *s-decomposable as a subgraph* if N can be obtained by gluing blocks by the rules in definition 1 and 5, and the resulting color of nodes in N ~~resulted by gluing of blocks must be compatible~~ coincides with the ~~original~~ color of N. (vertices

the same here

Remark 4. ~~First.~~ Note that if G is obtained by gluing a colored subgraph to a neighborhood of o, no edge in this neighborhood can be annihilated. ~~Second~~, For a given graph G and a selected node o, the set of neighborhoods of o in G, denoted by $\mathcal{N}_o$, ~~forms~~ is a partially ordered set by inclusion. We define three subsets of $\mathcal{N}_o$ as follows:

- $\mathcal{I}_o$ is the set of all decomposable neighborhoods each of which contains all edges incident to o.
- $\mathcal{D}_o$ is the set of all decomposable neighborhoods of o each of which is decomposable as a subgraph.
- $\mathcal{S}_o = \{N \subset \mathcal{I}_o \cap \mathcal{D}_o \mid N \text{ is minimal}\}$.

then, by definition,

~~By definition, for any considered node o,~~ if $\mathcal{S}_o$ is empty, the graph containing o is not s-decomposable.

Our goal is to find a combinatorial algorithm ~~to~~ which determines if a given graph is s-decomposable. According to remark 3, we can determine if a graph contains blocks from Table 2 by locating edges of weight 2 and analyzing their neighborhoods. We ~~discuss on~~ differ cases by the number of edges of weight 2 that are incident to the considered node o. Denote this number by n. According to the rules of gluing, n is at most 4 for any edge? in an s-decomposable graph. If o is not incident to any ~~of edges with~~ edge of weight two, we skip this node. Therefore, $n = 1, 2, 3$ or 4. Starting with any node o with $n = 4$, we check if $\mathcal{S}_o$ is non-empty by examining the following information that can be directly observed from the graph: degree of o, degree of the nodes that are connected to o by one edge, and the number and directions of the edges between node o and the nodes connected to o. If $\mathcal{S}_o$ is empty, the graph is not s-decomposable. If o is contained in a neighborhood in $\mathcal{S}_o$, we replace the neighborhood by another one which is *consistent* in the sense that the new graph is s-decomposable if and only if the original

maybe node?

one is. ~~We also want that~~ T the new neighborhood does not contain any edge ~~with~~ of weight 2. After all nodes with $n = 4$ are exhausted, we proceed to the nodes with $n = 3, 2, 1$ (in the s exact order). Finally, we get a graph that contains only edges of weight 1 and 4. The new graph is s-decomposable ~~iff~~ if and only if it is decomposable (see [1]). Then we apply the algorithm from [1] to determine it the graph is decomposable. Since all replacements are consistent, we can determine if the original graph is s-decomposable.

3 Algorithm

In [1], it is proved that we can assume that the graph is connected when only blocks in Table 1 are used. If the graph is s-decomposable, it is easy to see that we can make the same assumption as well. In fact, except ~~in~~ **II**, none of the edges can be annihilated by gluing a block from Table 2 to any graph. If **II**is glued to an existing graph, causing $\overline{uw}$ to be annihilated, nodes u, v are still connected via $\overline{uv}$ and $\overline{vw}$. Therefore, gluing new blocks will not break connectivity.

We collect all graphes obtained by gluing two blocks from Table 2, denoted by U, let: I don't understand what do you say here.

$$\mathcal{U} = U \cup \{\textbf{Ia,Ib,II,IIIa,IIIb,IV,V}\}$$

then in order to find $\mathcal{S}_o$ of a node o, it suffices to check if node o has a neighborhood that is isomorphic to ~~some~~ a graph in $\mathcal{U}$. By checking all nodes in $\mathcal{U}$, we analyze the results in Tables 3-8.

~~For~~ If any of the neighborhoods in $\mathcal{U}$, ~~if it~~ is a disjoint connected component (DCC), the algorithm ~~ends~~ stops. If the neighborhood may not be a disjoint connected component, we apply suitable replacement as suggested in the tables below. Note that we need those replacements to be consistent, i.e. the original graph is s-decomposable if and only if the new graph is. The consistency of all replacements can be checked by exhausting ~~the~~ analysis of all neighborhoods in $\mathcal{U}$ and Lemma 1 in [1].

3.1 $n = 4$

We have the following two situations:

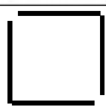
	A	B
Decomposition		
Degree of o	4	4
Replacement	~~Disjoint Connected Componenet~~ (DCC)	DCC

Table 3: $n = 4$

~~This case is trivial.~~

3.2 n=3

We have the following eight situations:

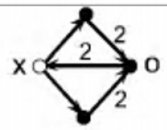
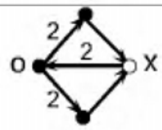
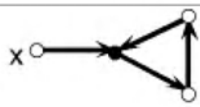
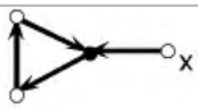
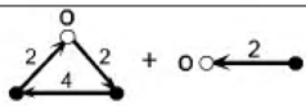
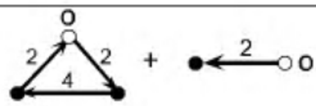
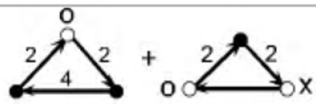
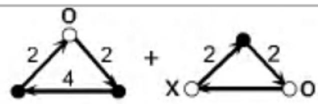
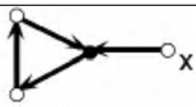
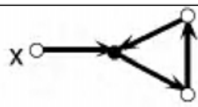
	A1	A2
Decomposition		
Degree of o	3	3
Replacement		
	B1	**B2**
Decomposition		
Degree of o	3	3
Replacement	DCC	DCC
	C1	**C2**
Decomposition		
Degree of o	4	4
Replacement		
	D1	**D2**
Decomposition		
Degree of o	5	5
Replacement	DCC	DCC

Table 4: n=3

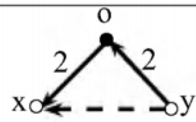
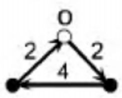
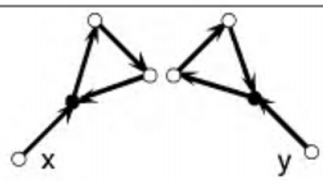
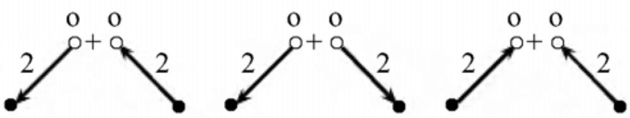
In this case the degree of the considered node can only be 3, 4 or 5, otherwise the graph is not s-decomposable. For a given graph G, to determine ~~which~~ neighborhood ~~o is contained in~~ of what type is considered, we examine the degree of o.

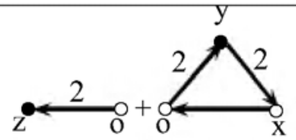
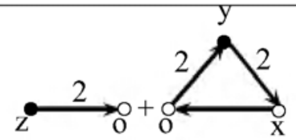
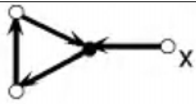
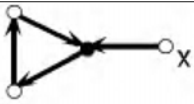
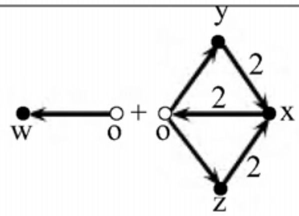
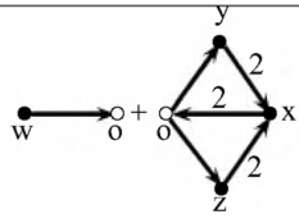
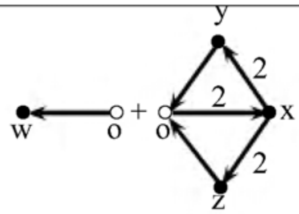
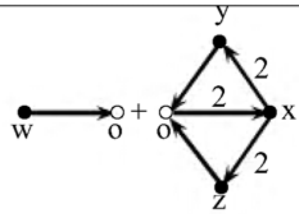
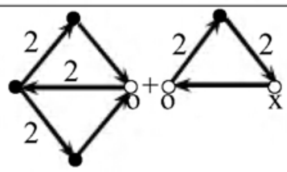
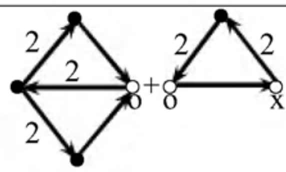
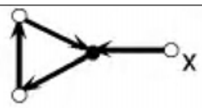
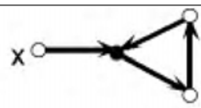
First suppose the degree of o is 3. We only need to consider **A1,A2,B1,B2**. We call the nodes connected to o by an edge *boundary nodes*. If one of the boundary nodes, denoted by x, is connected to the remaining two by edges of weight 1, o can only be contained in **A1** or **A2**. Note that in either cases, the degree of x is no less than 3, and the degrees of the remaining two boundary nodes are 2. If only two of the boundary nodes are connected, o can only be contained in **B1** or **B2**. In both cases, the neighborhoods are disjoint connected components.

Second, suppose the degree of o is 4. Node o can only be contained in **C1,C2**, otherwise the graph is not s-decomposable. Note that in this case, one boundary node is connected to o by an edge of weight 1. Denote this node by x. The remaining three boundary nodes are connected to o by edges of weight 2, two of them are connected by an edge of weight 4, the third one is connected to x by an edge of weight 2. Moreover, the degree of x is no less than 2, the remaining boundary nodes all have degree 2.
Finally, suppose the degree of o is 5. In this case, o can only be contained in **D1,D2**, otherwise the graph is not s-decomposable. In either case, the neighborhood is a disjoint connected component.

3.3 $n = 2$

We have the following 13 situations:

	A	**B**
Decomposition		
Degree of o	2	≥ 2
Replacement		
	C	
Decomposition		
Degree of o	2	
Replacement	DCC	
	D1	**D2**

Decomposition		
Degree of o	3	3
Replacement		
	E1	**E2**
Decomposition		
Degree of o	4	4
Replacement	DCC	DCC
	E3	**E4**
Decomposition		
Degree of o	4	4
Replacement	DCC	DCC
	F1	**F2**
Decomposition		
Degree of o	5	5
Replacement		
	F3	**F4**

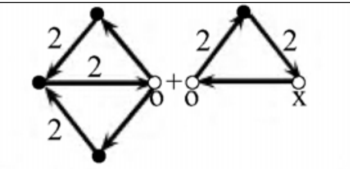
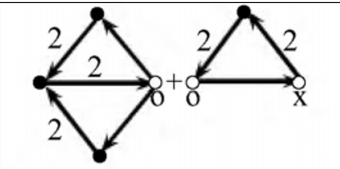
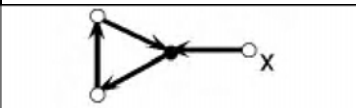
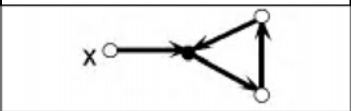
Decomposition	 	
Degree of o	5	5
Replacement	 	

Table 5: n=2

To determine if the considered node o is contained in any of the neighborhoods in Table 5, we ~~discuss~~ consider the degree of o. Note that when $n = 2$, the degree of ~~the considered~~ node o ~~can~~ in Table 5 takes only ~~be~~ values $2, 3, 4$ or 5 ~~by Table~~ 5, otherwise, the graph is not s-decomposable.

If the degree of o is 2, o in s-decomposable graph can have only ~~can only be contained in~~ a neighborhoods of types ~~as in cases~~ **A**,**B**,**C**, (see Table 5) otherwise the graph is not s-decomposable. In these cases we Denote the other endpoints of the edges of weight ~~two~~ 2 by x, y.

- If x, y are ~~dis~~ not connected by an edge, there are two possibilities. First, o can be obtained in one of the neighborhoods as shown in case **C**; second, the neighborhood can be obtained from annihilating edge $\overleftarrow{xy}$. To determine how the neighborhood is obtained, note that in the first case, the neighborhood is a disjoint connected component, and in the second case, the edge $\overleftarrow{xy}$ can be annihilated by an edge from a spike, a triangle or the mid-edge of a diamond. Hence if the degrees of nodes x, y are both 2, and are both directed away or from node o, then the neighborhood is obtained in the way shown in the second or third picture in case **C**. If two edges have different orientations, there are two possible decompositions. If the degree of nodes x, y are both greater than 2, the graph is decomposable only if the neighborhood is obtained by gluing another block to the one shown in the case **A**.

I did not understand what do you want to say. Please, explain.

- If x, y are connected by an edge of weight 4 from y to x, there are two cases. First, the neighborhood can be obtained from gluing an edge $\overrightarrow{yx}$ to the graph as in case **A**. In this situation, the degrees of nodes x, y are at least 2; Second, the neighborhood can be obtained from the ~~one in~~ case **B**. Therefore, ~~in this case,~~ to distinguish the above two cases, we first check the degrees of o. If o has degree greater than 2, o is contained in a neighborhood as shown in case **B**. If o has degree 2, we check the degrees of x, y: if the degrees of both x, y are two, the neighborhood is a disjoint connected component and has two possible decompositions; if the degrees of both x, y exceed 2, o must be contained in a neighborhood ~~as shown in~~ **A**. In latter case, after applying the corresponding replacement, we ~~need to~~ keep edge $\overline{xy}$ and change its weight from 4 to 1.

- If x, y are connected by an edge of weight one from y to x, the neighborhood can only be obtained from the ~~one in~~ case **A**.

Next suppose the degree of o is 3. In this case node o can only be contained in a neighborhood shown in **B**,**D1** or **D2**, otherwise the graph is not s-decomposable. To distinguish these cases First we check if the nodes connected to o by an edge of weight 2 are connected by an edge of weight 4. If so, o must be contained in **B**. If not, we denote the three nodes connected to o by x, y, z, where y, z are connected to o by edges with weight 2. Note that x must be connected to o via an edge with weight 1, x, y must be connected via an edge $\overrightarrow{yx}$ with weight 2, degrees of y and z must be 2 otherwise the graph is not s-decomposable. Node o is contained in a neighborhood shown as in **D1** if o, z are connected via an edge $\overrightarrow{oz}$, **D2** if via $\overrightarrow{zo}$.

Next suppose the degree of o is 4. In this case node o can only be contained in a neighborhood shown in one of **B**,**E1**-**E4**, otherwise the graph is non s-decomposable. By the same argument as in this previous case, we check if o is contained in **B**. If not, we need to determine if o is contained in any of ~~these neighborhoods~~ E1-E4. Denote the nodes that are connected to o by edges with weight 1 by y, z, the nodes that are connected to o by edges with weight 2 by x, w. Then y, z must both be connected to one of the nodes that are connected to o. Assume y, z are both connected to x, then $\overline{xy}, \overline{xz}$ both have weight 2. ~~Judging from picture~~ By **E1**-**E4**, $\overline{ow}$ must have weight 2, and the degree of w is 2. ~~We can~~ By examin~~e~~ation of the orientation~~s~~ of the edges incident to o ~~to~~ we determine ~~with~~ in which type of neighborhood o is contained ~~in~~.

Finally, if the degree of o is 5, ~~o~~ it can only ~~be~~ be contained in a neighborhood shown in one of **B**,**F1**-**F4**, otherwise the graph is non s-decomposable. ~~Similarly~~ As above, we check if o is contained in **B**. If not, we denote the boundary nodes of o by x, y, z, u, v, where x, y, z are connected to o by edges of weight 1, u, v are connected to o by edges of weight 2. Note that if the graph is s-decomposable, one of u, v must be connected to two nodes among x, y, z by edges of weight 2. Assume u is connected to y and w by edges of weight 2. Then, v must be connected to x by another edge of weight 2, and $\deg(y) = \deg(w) = \deg(v) = 2$, $\deg(u) = 3$, $\deg(x) \geq 2$.

3.4 $n = 1$

In this case, there is only one edge with weight 2 that is incident to o. Denote the other endpoint of ~~such~~ this edge by p. We consider the number of edges incident to p with weight 2. Denote this number by m.

If $m = 1$, there are two cases, as shown Table. 6. We can only attach blocks containing no edge of weight two to the node p. In both cases, degree of o is one. It is easy to determine if o is contained in **A1** or **A2**.

	A1	**A2**
Decomposition	$p \xleftarrow{2} o$	$p \xrightarrow{2} o$

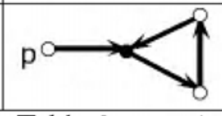
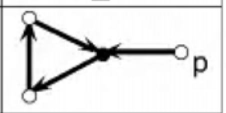
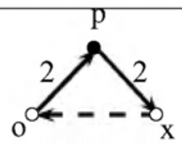
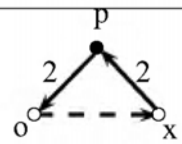
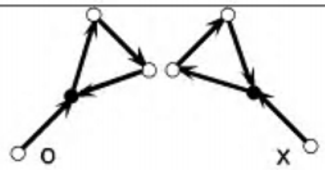
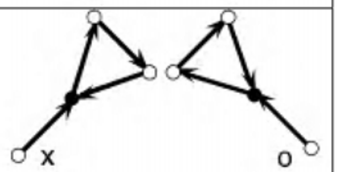
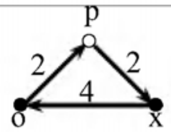
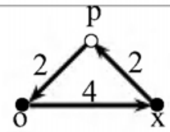
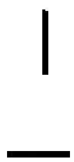
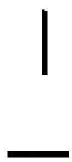
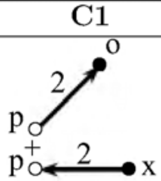
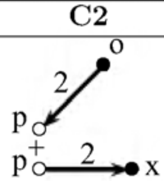
Degree of p	≥ 1	≥ 1
Replacement		

Table 6: $m = 1$

If $m = 2$, there are ten possible cases, as shown in Table 7. In either of the cases **A1**,**A2**, we can only attach to o blocks containing no edge of weight 2. Hence after applying the corresponding replacement, there is no edge with weight 2 that is incident to o. In case **B1** or **B2**, we can only attach to p blocks containing no edge of weight 2.

	A1	**A2**
Decomposition		
Degree of p	2	2
Replacement		
	B1	**B2**
Decomposition		
Degree of p	≥ 2	≥ 2
Replacement		
	C1	**C2**
Decomposition		
Degree of p	2	2

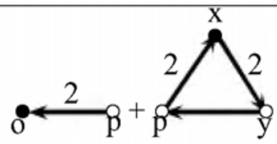
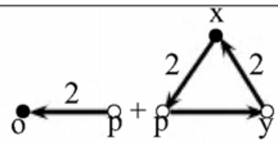
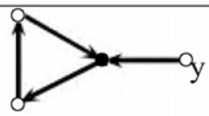
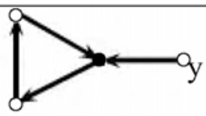
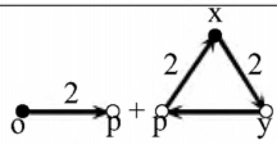
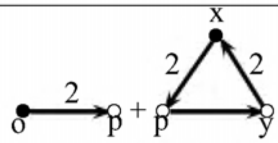
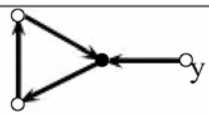
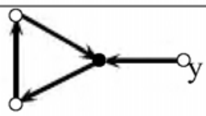
Replacement	DCC	DCC
	D1	**D2**
Decomposition		
Degree of p	3	3
Replacement		
	D3	**D4**
Decomposition		
Degree of p	3	3
Replacement		

Table 7: $m = 2$

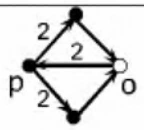
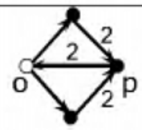
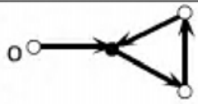
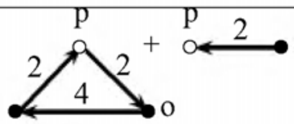
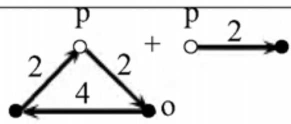
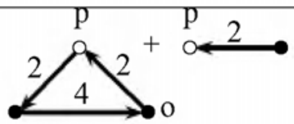
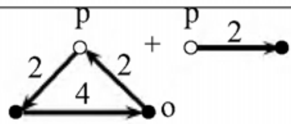
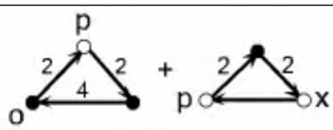
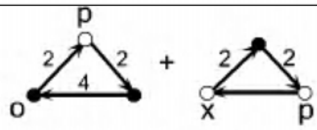
Table 7 gives all possible cases ~~when~~ with $m = 2$. To determine ~~which~~ the type of neighborhood o is contained in, let us denote the node connected to o by an edge ~~with~~ of weight 2 by p, then examine the degree of p. According to Table 7, if the graph is s-decomposable, $\deg(p) \geq 2$.

Suppose the degree of p is 2, we denote the other node that is connected to p by x. Note that the weight of $\overline{px}$ must be 2. If x is connected to o by an edge with weight 4, then o must be contained in ~~a~~ neighborhood ~~as shown in~~ **B1** or **B2** depending on the orientation of edges. Note that in this case, the graph is a disjoint connected component. If $\overline{xo}$ has weight 1, then o is contained in ~~a~~ neighborhood ~~as shown in~~ **A1** or **A2** depending on the orientation of edges. If x is not connected to o, then the graph must be a disjoint connected component ~~as shown in~~ **C1** or **C2**.

Next, suppose the degree of p is 3. By Table 7, there are two edges with weight 2 that are incident to p, one of which is $\overline{op}$. Denote the other edge ~~with~~ of weight 2 by $\overline{ox}$ ~~with~~ (the other endpoint is x). If x is connected to o by an edge ~~with~~ of weight 4, then o is contained in ~~a~~ neighborhood ~~as shown in~~ **B1** or **B2**. If x is not connected to o, then o ~~must be contained~~ lies in ~~a~~ neighborhood ~~as shown in one of~~ **D1**-**D4**. Note that in the latter case, the degree of o is 1. If it is neither of the above two situation, the graph is not s-decomposable.

Finally if the degree of p is greater than 3, o must be contained in ~~a~~ neighborhood ~~as shown in~~ **B1** or **B2**.

If $m = 3$, there are fourteen cases, as shown in Table.8.

	A1	**A2**
Decomposition		
Degree of p	3	3
Replacement		
	B1	**B2**
Decomposition		
Degree of p	3	3
Replacement	DCC	DCC
	B3	**B4**
Decomposition		
Degree of p	3	3
Replacement	DCC	DCC
	C1	**C2**
Decomposition		
Degree of p	4	4
Replacement		
	C3	**C4**

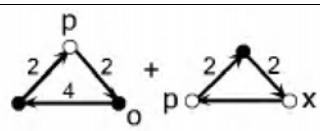
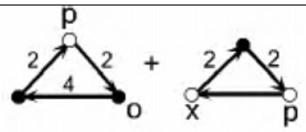
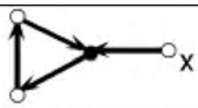
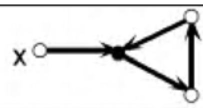
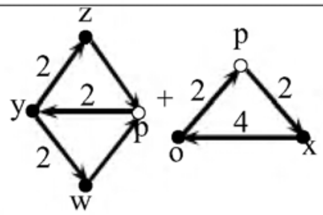
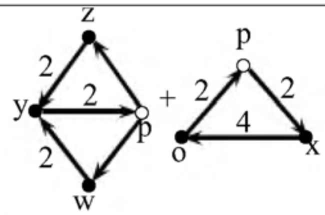
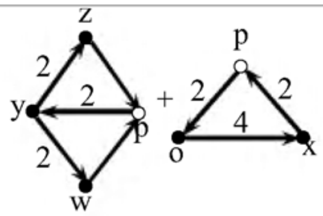
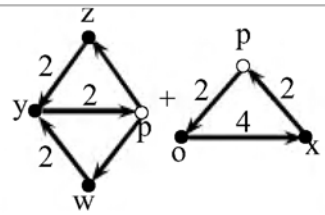
Decomposition		
Degree of p	4	4
Replacement		
	D1	**D2**
Decomposition		
Degree of p	5	5
Replacement	DCC	DCC
	D3	**D4**
Decomposition		
Degree of p	5	5
Replacement	DCC	DCC

Table 8: m=3

Note that the degree of node o in all pictures is 2 except ~~in~~ **A1** and **A2**. Therefore, if the considered node has degree larger than 2, it can only be contained in ~~the~~ neighborhood ~~shown in~~ **A1** or **A2**. In both ~~neighborhoods~~ cases, there are two nodes, denoted by x, y, that are connected to o by edges ~~with~~ of weight 1, and p is connected to both ~~nodes~~ x and y by edges ~~with~~ of weight 2. Moreover, the degree of p is 3, the degrees of x, y are both 2. Suppose ~~the considered node~~ o has degree 2. ~~We consider the degree of p~~. According to Table 8, $\deg(p) = 3, 4$ or 5.

First, suppose $\deg(p) = 3$. o can only be contained in ~~a~~ neighborhood ~~shown in~~ **B1,B2,B3** or **B4**. Note that in all these cases, all edges incident to p has weight 2, and the graph is a disjoint connected component.

Next, suppose $\deg(p) = 4$. Then o can only be contained in a neighborhoods ~~shown in~~ of type **C1,C2,C3** or **C4**. In all these cases, p is incident to four edges of weight 2. Also p is connected to a node which is also connected to o by an edge ~~with~~ of weight 4. Among the four nodes connected to p, three of them, including o, have degree 2, the remaining node has degree no less than 2. We can check the orientations of all edges to determine which neighborhood o is contained in.

Finally, suppose $\deg(p) = 5$. In this case, o can only be contained in ~~a~~ neighborhood ~~shown in~~ **D1-D4**, and the graph is a disjoint connected component. In all these cases, p is incident to three edges of weight 2. Denote the other endpoints of these edges besides o by ~~o,~~ x, and y ~~(o is the considered node)~~. Node p is also incident to two edges ~~with~~ of weight 1. Denote the other endpoints of these two edges by z, w. According to Table 8, z, w must both be connected to one of x, y by edges of weight 2. Assume it is y. Then o is connected to x by an edge of weight 4. Note that in this case $\deg(y) = 3$, $\deg(z) = \deg(w) = \deg(x) = 2$. We can determine which neighborhood o is contained in by examining the orientations of the edges.

4 Summary

In Section 3, we exhausted all nodes that are incident to some edges with weight 2. We also replace a neighborhood of any such node by a consistent one which does not contain any edge with weight 2. Therefore, for any given weighted graph, we can determine if it is s-decomposable, and simplify it into a graph containing only edges with weight 1 or 4. Then we apply the algorithm in [1] to determine if it is block decomposable. Note that every node is examined at most twice: once in the procedure as in 3, once in the algorithm in [1]. Hence the algorithm is linear in the size of the given graph.

Apply the algorithm to Theorem 3, we get the following corollary:

Corollary. *Given a skew-symmetrizable matrix B, there exists an algorithm linear in the size of B to determine if B has finite mutation type.*

Proof. Assume the size of B is no less than 3. First, we check if B is mutation-equivalent to one of the seven exceptional types in Theorem 3. If so, B is mutation finite. Since the sizes of all seven types do not exceed 6, it only takes finite number of operation ~~step to finish this examination~~. If none of the seven types is mutation equivalent to B, we apply our algorithm to the associated adjacency graph of B. By the previous argument, the number of operation it requires is linear in the size of B. If the adjacency graph is confirmed to be s-decomposable, B has finite mutation type. □

Remark 5. If diagram G is s-decomposable, our algorithm can recover the blocks used to obtain G since every step of replacement is consistent. In particular, we can determine the ideal tagged triangulation of bordered surfaces with marked points to each decomposition.

Remark 6. A connected diagram G has non-unique decomposition ~~iff~~ if and only if G is isomorphic to one of the two diagrams in Figure 3.

(it is a bad taste to write "iff" in papers)

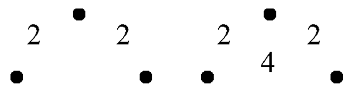


Figure 3

References

[1] W. Gu. A decomposition algorithm for the oriented adjacency graph of the triangulations of a bordered surface with marked points.

[2] Anna Felikson, Michael Shapiro, and P. Tumarkin. Cluster algebras of finite mutation type via unfoldings. *arXiv:1006.4276v3.*

[3] A. Felikson, M. Shapiro, and P. Tumarkin. Skew-symmetric cluster algebras of finite mutation type. *arXiv:0811.1703.*

[4] G. Lusztig. Introduction to quantum groups. *Progr. Math. Vol. 110, Birkhauser, Boston, 1993.*