ALCUIN NUMBERS

ADI ADVANI

Abstract. This thesis explores the Alcuin Number problem. Our goal is to categorize all Small-Boat graphs for a given vertex cover number $\tau(G) = n \in \mathbb{N}$. We attempt this by developing a family of Small-Boat Subgraphs (SBSs), which are configurations that if present in a graph G ensure that G is small boat. We provide the complete list of all such SBSs for $|\tau(G)| = 1$, $|\tau(G)| = 2$ and $|\tau(G)| = 3$ and also provide a method to find SBSs for larger $\tau(G)$ by combining smaller SBSs.

Contents

Date: 2024-04-12.

This document is a senior thesis submitted to the Department of Mathematics and Statistics at Haverford College in partial fulfillment of the requirements for a major in Mathematics.

1. Introduction

You must cross a river in your boat, which can hold up to one other entity (we say the boat has "size 1"). You have a fox, a goat and a piece of cabbage that must all make it to the other bank. If you leave the fox with the goat, the goat is eaten. If you leave the cabbage with the goat, the cabbage is eaten. [5, p. 1]

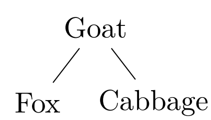
The solution is as follows.

- First you must transport the goat.
- Then, return to the first bank, take the fox to the second bank.
- Since you cannot leave the fox and the goat together, you must take the goat back to the first bank.
- Then, take the cabbage to the second bank, leaving the goat on the first bank.
- Finally, return one last time for the goat.

Now consider the possibility, what if you had two foxes? This problem is now unsolvable with a boat of size 1. This is because you are now unable to leave the goat on the first bank to pick up the cabbage, as it would be left with the second fox (Try it!). In this case, you need a boat of size 2 to solve the riddle.[1]

The small-boat/large boat problem is a generalization of this riddle that asks a simple question. For any given number of animals, and any possible set of interactions between them, what is the smallest boat size required to transport them across?

Often, these problems are represented using graphs [2, p. 1], where the vertices represent animals, and an interaction (one animal eating another) is represented by an edge connecting the two vertices. For example, the original fox-goat-cabbage problem would be represented by the following graph.



Note that to us, being eaten or eating another animal is identical mathematically speaking, since we are attempting to avoid any interactions at all. So, a fox-goat-cabbage graph can be looked at as a fox-goat-fox graph instead! Thus, all graphs are undirected. In addition to this, all graphs are simple, which means there are no loops or multiple edges between vertices, as an animal will never eat itself and an animal cannot eat another multiple times.

[1] With a boat of size 2, it is quite easy to solve. You can keep the goat in the boat and transport the foxes and cabbage over one by one.

2. Background and Definitions

2.1. **The Alcuin Number Problem.** This thesis aims to find the Alcuin Number for any simple, undirected graph G. The Alcuin Number ALCUIN(G) is equal to the smallest possible boat size necessary to transfer all the vertices (animals) across a river safely (by avoiding the interactions represented by edges in G).

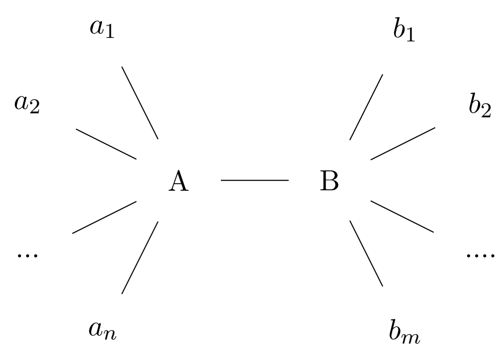
Definition 2.1. [1, p. 97] [3, p. 317] *A vertex cover of a graph* $G = (V, E)$ is a set of vertices $V' \subset V$ where every single edge in G has an end vertex in V'. *The vertex cover number of* G is the number of vertices in a smallest possible (minimal) vertex cover of G.

Notation: The vertex cover number of a graph G is written as $|\tau(G)|$ where $\tau(G)$ is a minimal vertex cover.[2]

In general, we assume from this point onwards that any vertex cover we consider is minimal.

A vertex cover of the graph is extremely important to us, as it provides us with a smallest set of vertices with the property that every edge in the graph is adjacent to at least one of the vertices in the chosen set. Later we will see that $\tau(G)$ is the lower bound on the boat size (Proposition 2.4).

Example 2.2. Consider the following graph, where $n, m \in \mathbb{N}$.



A minimal vertex cover of this graph consists of the two elements A and B, which means that $|\tau(G)| = 2$. This is because when A and B are removed from the graph, no edges are left. In fact, $\{A, B\}$ is the unique minimal vertex cover in this case.

Definition 2.3. [2, Lem 1]. *The Alcuin number of a graph,* ALCUIN(G) is the smallest possible boat size required to solve the Alcuin River Crossing problem on G.

In general, it is our goal to calculate ALCUIN(G) for any graph $G = (V, E)$, so let's find an upper and lower bound for this quantity.

---

[2] There may be more than one minimal vertex cover. See Lemma 3.2.

Proposition 2.4 ([5, Lem 2.1])**.** $\textsc{Alcuin}(G) \geq |\tau(G)|$.

Proof. To start the solution to the Alcuin problem for G, we must load into the boat all vertices in some minimal vertex cover $\tau(G)$. Otherwise, we leave an edge on the first bank after the first step, breaking the rules of the problem. Therefore, $\textsc{Alcuin}(G) \geq |\tau(G)|$. □

Small boat graphs are graphs G where $\textsc{Alcuin}(G) = |\tau(G)|$.

Proposition 2.5 ([5, Lem 2.1])**.** $\textsc{Alcuin}(G) \leq |\tau(G)| + 1$.

Proof. It is always possible to solve any graph with a boat of size $\textsc{Alcuin}(G) = |\tau(G)|+1$. This is because we can put the entire minimal vertex cover on the boat and use the one extra spot to transfer the rest of the elements over. □

Large boat graphs are graphs G where $\textsc{Alcuin}(G) = |\tau(G)| + 1$.

In the propositions above, we have shown that $|\tau(G)| \leq \textsc{Alcuin}(G) \leq |\tau(G)|+1$. So, we see that for any graph G there are only two possible values for $\textsc{Alcuin}(G)$ i.e, G is either a *small boat graph* or a *large boat graph*. This simplifies things greatly- assuming we know the vertex cover of the graph (which is usually hard to find). Csorba remarks that, in general, finding the vertex cover of a graph is an NP-Complete problem [2, p. 4].

Now, our problem can be succinctly stated as: **For any given graph G, is G a small boat or a large boat graph?**

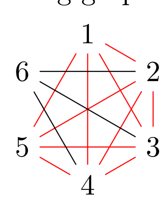
Example 2.6. Consider the graph in Example 2.2 above. Since its vertex cover number is $|\tau(G)| = 2$, we know that if it is a small boat graph, it has a minimum boat size of 2, and if it is a large boat graph, it has a minimum boat size of 3. We find out which type it is later in the section (Example 2.11).

Now we move into some graph theoretic definitions that prove useful in later lemmas. These quantities are related to the vertex cover of a graph and simplify our solutions down the line.

2.2. **Graph Theory Background.**

Definition 2.7. [3, p 27] For $n \in \mathbb{N}$, the *clique* K_n is the graph G on n vertices that is *complete*. By this we mean that every vertex is connected to every other vertex in G. *The clique number* $|\mathbb{C}| \in \mathbb{N}$ *of* G is the size of the largest clique in G.

Example 2.8. Consider the following graph G.



In this case, vertices $\{1, 2, 3, 4, 5\}$ form a clique K_5 (marked in red) as a subgraph of G. This is clearly the largest clique in G so the clique number $|\mathbb{C}|$ is 5.

Definition 2.9. [3, p. 296] Let $G = (V, E)$ be a graph. *A stable set* V' is a set of vertices where there are no edges between the elements of V'. *The anti-clique number* $|\mathbb{A}|$ of a graph G is the size of a largest stable set $\mathbb{A} \subset G$.

Remark 2.10. Given a graph G with a vertex cover $\tau(G)$, we fix $\mathbb{A} \subset \tau(G)$.

In Example 3.8 there are 2 largest stable sets $\{1, 6\}$ and $\{1, 5\}$ (each with size 2). So, $|\mathbb{A}| = 2$.

The proof of the following proposition is straightforward.

Proposition 2.11. *When* $\tau(G)$ *is a vertex cover (even if it is not minimal), note that the set complement of* $\tau(G)$ *is an stable set which we write as* $\overline{\tau(G)}$. *Also, the complement of a stable set is a vertex cover (though it is not necessarily minimal)*

Finally, we introduce some notation for "solutions" to the Alcuin Problem for a given graph G. This notation is essential in most proofs below.

Definition 2.12. [2, Section 3] A *schedule* with boat size b for a graph $G = (V, E)$ is defined as a finite sequence of triples:

$$
\begin{array}{llll}
(1) & L_1 & B_1 & R_1 \\
(2) & L_2 & B_2 & R_2 \\
(\vdots) & \vdots & \vdots & \vdots \\
(s) & L_s & B_s & R_s
\end{array}
$$

The elements of these triples are subsets of V that satisfy the 3 conditions below. The *length* of the schedule is the number of triples present in the schedule. All schedules have odd length.

- For every k, the sets L_k, B_k, R_k form a partition of V. The sets L_k and R_k are stable sets in G. The set B_k contains at most b elements.
- The sequence starts with $L_1 = V$, $B_1 = \emptyset$ and $R_1 = \emptyset$, and the sequence ends with $L_s = \emptyset$ and $B_s = \emptyset$, and $R_s = V$.
- For even $k \geq 2$, we have $B_k \cup R_k = B_{k-1} \cup R_{k-1}$ and $L_k = L_{k-1}$. For odd $k \geq 3$, we have $L_k \cup B_k = L_{k-1} \cup B_{k-1}$ and $R_k = R_{k-1}$.

R_k represents the vertices on the right bank, B_k the vertices on the boat and L_k the vertices on the left bank.

Example 2.13. Consider the graph in Example 2.2 above. It is a small boat graph. This means we can solve it using a boat of size 2. Consider the following schedule.

(1)	$A, B, a_1, a_2, ..., a_n, b_1, b_2, ..., b_m$	$\emptyset$	$\emptyset$
(2)	$a_1, a_2, ..., a_n, b_1, b_2, ..., b_m$	A, B	$\emptyset$
(3)	$a_1, a_2, ..., a_n, b_1, b_2, ..., b_m$	B	A

Now, we transfer over $b_1, b_2, ..., b_m$ one by one using the empty spot on the boat:

$(\vdots)$	$\vdots$	$\vdots$	$\vdots$
$(2m+3)$	$a_1, a_2, ..., a_n$	B	$A, b_1, b_2, ..., b_m$
$(2m+4)$	$a_2, ..., a_n$	B, a_1	$A, b_1, b_2, ..., b_m$
$(2m+5)$	$a_2, ..., a_n$	A, B	$a_1, b_1, b_2, ..., b_m$
$(2m+6)$	$B, a_2, ..., a_n$	A	$a_1, b_1, b_2, ..., b_m$

Now we transfer over $a_2, ..., a_n$ one by one using the empty spot on the boat:

$(\vdots)$	$\vdots$	$\vdots$	$\vdots$
$(2m+2n+7)$	B	A	$a_1, a_2, ..., a_n, b_1, b_2, ..., b_m$
$(2m+2n+8)$	$\emptyset$	A, B	$a_1, a_2, ..., a_n, b_1, b_2, ..., b_m$
$(2m+2n+9)$	$\emptyset$	$\emptyset$	$A, B, a_1, a_2, ..., a_n, b_1, b_2, ..., b_m$

Therefore, the graph in Example 2.2 is in fact, small boat and solvable with boat size $2 = |\tau(G)|$.

Now that we have some notation and background on the topic, we can dive into some short proofs and lemmas relating to the Alcuin numbers.

3. Schedules and Properties

3.1. **Representing a simple graph G.** Fix a minimal vertex cover $\tau(G)$. Every connected simple graph G can be broken down into its vertex cover $\tau(G)$ and the stable set $\overline{\tau(G)}$. Each vertex in $\overline{\tau(G)}$ is adjacent to at least one vertex in $\tau(G)$.

Example 3.1. Consider the same example graph from Example 2.2 above. We have the following:

- $\tau(G) = \{A, B\}$
- $\overline{\tau(G)} = \{a_1, ..., a_n, b_1, ..., b_m\}$ which is a stable set as each vertex in $\overline{\tau(G)}$ only connects to the vertices in $\tau(G)$. In this case each vertex in $\overline{\tau(G)}$ is adjacent to exactly one vertex in $\tau(G)$.

3.2. **Reversible Schedules.** A simple observation about schedules, is that if a schedule works to transport a graph G from the left bank to the right bank, it can always work backwards to transport the same graph G from the right bank to the left [2, Lem 2]. Therefore, all schedules are reversible.

3.3. **Non-unique minimal vertex covers.**

Lemma 3.2. *[2, Lem 4.3]*[3] *Any graph G with more than one minimal vertex cover is a small boat graph.*

Proof. Let $G = (V, E)$ have two minimal vertex covers $\tau(G)$ and $\tau'(G)$ each of size n. Then, $\overline{\tau(G)}$ and $\overline{\tau'(G)}$ form stable sets each of size $|V| - n$. Define the following sets.

- $X_1 = \overline{\tau(G)} \cap \overline{\tau'(G)}$
- $X_2 = \overline{\tau(G)} \cap \tau'(G)$
- $Y = \overline{\tau'(G)} \cap \tau(G)$

Note the following:

(1) $X_1 \cup X_2 = \overline{\tau(G)}$, where $X_1 \cap X_2 = \emptyset$.
(2) $X_1 \cup Y = \overline{\tau'(G)}$, where $X_1 \cap Y = \emptyset$.
(3) We also know that $Y \neq \emptyset$ and $X_2 \neq \emptyset$ as $\tau(G) \neq \tau'(G)$.

We can now propose a schedule with boat size $\tau(G) = n$.

$$\begin{array}{llll} (1) & G & \emptyset & \emptyset \\ (2) & \overline{\tau(G)} & \tau(G) & \emptyset \\ (3) & \overline{\tau(G)} & \tau(G) - Y & Y \end{array}$$

It is possible to leave Y on the right bank because $Y \subset \overline{\tau'(G)}$, which is a stable set.

Now, there are exactly $|Y|$ slots empty on the boat. We now break X_1 into $|Y|$ sized sets and transport it across to the other bank. This can be done since X_1 is the intersection of two stable sets and is therefore a stable set. In fact, $X_1 \cup Y$ is a stable set.

Note that since $\overline{\tau(G)} - X_1 = X_2$, we are left with X_2 on the first bank.

$$\begin{array}{llll} (4) & \overline{\tau(G)} & \tau(G) - Y & Y \\ & \vdots & & \\ (5) & \overline{\tau(G)} - X_1 & \tau(G) - Y & Y \cup X_1 \end{array}$$

[3] I have arrived at this proof independently, but [2] has a different proof of this result, which looks quite different from the one below.

Note that since $\overline{\tau(G)} - X_1 = X_2$, we are left with X_2 on the left bank and that the right bank now holds $Y \cup X_1 = \overline{\tau'(G)}$.

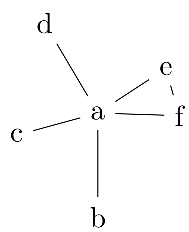
$$
\begin{array}{llll}
(5) & X_2 & \tau(G) - Y & \overline{\tau'(G)}
\end{array}
$$

At the start of the schedule, we fixed that $X_2 = \overline{\tau(G)} - \overline{\tau'(G)}$ and $Y = \overline{\tau'(G)} - \overline{\tau(G)}$. Also, we know that $|\tau(G)| = |\tau'(G)|$. Using these two facts, we can deduce that $|X_2| = |Y|$. So, using the $|Y|$ spots on the boat, we can now transfer over X_2.

$$
\begin{array}{lcrr}
(6) & X_2 & \tau(G) - Y & Y \cup X_1 \\
(7) & \emptyset & (\tau(G) - Y) \cup X_2 & Y \cup X_1 \\
(8) & \emptyset & \emptyset & G
\end{array}
$$

So, we have shown that for any graph G with two vertex covers $\tau(G)$ and $\tau'(G)$, it is always a small boat graph. □

Example 3.3. Consider the following graph, which has two minimal vertex covers, $\{a, e\}$ and $\{a, f\}$. So, in the notation of the proof we have $X_1 = \{b, c, d\}$, $X_2 = \{f\}$ and $Y = \{e\}$.



Consider the following schedule (with boat size $|\tau(G)| = 2$), which exactly follows the proof of Lemma 3.2:

$$
\begin{array}{lrrr}
(1) & a, b, c, d, e, f & \emptyset & \emptyset \\
(2) & b, c, d, f & a, e & \emptyset \\
(3) & b, c, d, f & a & e \\
 & \vdots & \vdots & \vdots \\
(4) & f & a & e, b, c, d \\
(5) & \emptyset & a, f & e, b, c, g \\
(6) & \emptyset & \emptyset & a, f, e, b, c, g
\end{array}
$$

Thus, this graph with two minimal vertex covers is small boat.

Lemma 3.2 is useful, because it reduces the number of graphs we are to consider significantly. In fact, it leaves only one interesting case: when $\tau(G)$ is minimal and unique.

Remark 3.4. From now on, we will assume that for any graph G, $\tau(G)$ is minimal and unique.

3.4. **The second and second-last move in a schedule.**

Lemma 3.5. *The second and second-last B_2 and B_{s-1} in a schedule are the same*

For any small boat graph G with a unique minimal vertex cover $\tau(G)$ and any viable schedule S, B_2 and B_{s-1} are both equal to $\tau(G)$.[4]

Proof. Recall from Remark 3.4 that we are considering graphs with unique vertex covers only.

Notice that every sequence in a schedule starts with $B_2 = \tau(G)$; otherwise, L_2 is not an stable set. Now, we show that $B_{s-1} = \tau(G)$ (and $L_{s-1} = \emptyset$ naturally).

Note the following important information:

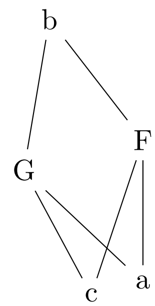
(1) R_{s-1} is an stable set since the boat is approaching the right bank for the last time.
(2) We must have $L_{s-1} = \emptyset$, and $B_{s-1} \cup R_{s-1} = V$.
(3) The maximum possible size of B_{s-1} is $|\tau(G)|$ as we assume G is a small boat graph.

So, the vertex cover B_{s-1} must have size equal to $|\tau(G)|$. By hypothesis $\tau(G)$ is unique which means that $\tau(G) = B_{s-1}$ as desired. $\square$

This lemma is useful as it provides us with information about how the vertices in the vertex cover must move during any schedule. We know that the first and last move must be $\tau(G)$, which means that any vertices in the vertex cover that are transferred to the right bank during the schedule must eventually be transferred back before the 2nd last step.

Example 3.6. Consider the following graph.

[4] This is an original proof.



This graph is small boat, and has vertex cover $\{F, G\}$. Consider the following schedule ($|\tau(G)| = 2$):

(1)	a, b, c, F, G	$\emptyset$	$\emptyset$
(2)	a, b, c	**F,G**	$\emptyset$
(3)	a, b, c	$\emptyset$	F, G
(4)	c	a, b	F, G
(5)	c	F, G	a, b
(6)	G	F, c	a, b
(7)	G	F	a, b, c
(8)	$\emptyset$	**F,G**	a, b, c
(9)	$\emptyset$	$\emptyset$	a, b, c, F, G

This schedule has its first and last boat $B_2 = B_8 = \{F, G\}$, as expected.

3.5. **The third move in a schedule.**

Lemma 3.7. *For a given small boat graph G, there is always a viable schedule with $B_2 = \tau(G)$ and $R_3 = \mathbb{A}$*[5]*. Where $\mathbb{A}$ is define in Remark 2.10.*

Proof. Once you have the vertex cover on the boat ($B_2 = \tau(G)$), you can choose to drop a subset of the maximal anticlique off on the right bank, or the entire anticlique $\mathbb{A}$ ($R_3 = \mathbb{A}$). Any viable schedule must first involve dropping off some subset of the anticlique $P \subseteq \mathbb{A}$. In this case, we will show that we can provide an alternate schedule that has the exact same length where the only difference is that we drop off the entire anticlique off so that $R_3 = \mathbb{A}$.

Here is the modified schedule.

- First, pick up the vertex cover so that $B_2 = \tau(G)$, then drop off the anticlique in step 3 so that $R_3 = \mathbb{A}$. The change is that in the original

[5]This is an original proof.

schedule we drop off $R_3 = P \subseteq \mathbb{A}$ so we have $|P| < |\mathbb{A}|$ spots on our boat for step 4, but in the modified schedule we have $|\mathbb{A}|$ spots on our boat.

- Now, in Step 4, we may assume that the original schedule picks up some vertices from the left bank. We will proceed with the same Step 5 as the original schedule which is always possible as we have strictly more spots on our boat ($|P| < |\mathbb{A}|$).
- In Step 5, we can drop these elements off on the right bank[6], and move the elements $\mathbb{A} - P$ back onto the boat. This is again always possible since after dropping elements on the right bank we have $|\mathbb{A}|$ slots on the boat, and because our original schedule is viable we know that moving $\mathbb{A} - P$ to the boat is sure to leave any conflicting vertices on the right bank.

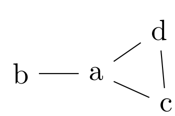
After this point, we follow the remaining steps exactly as in the original schedule, and we have not changed the length, only changed the first few steps. Thus, it is always possible to have a schedule with $B_2 = \tau(G)$ be followed with $R_3 = \mathbb{A}$. □

4. Small-Boat Configurations

Robertson and Seymour showed that for every "minor-closed" family of graphs, there is a finite set of "forbidden minors" [4]. This theorem is quite powerful— as it can be applied to many different concepts in graph theory. For example, we can categorize any graph G as being planar/non-planar or linked/non-linked.

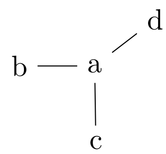
Following Robertson's and Seymour's methods, the goal of this chapter is to develop a family of subgraphs that inform us if a graph is small boat or large boat. Note that since removing an edge or contracting an edge can change the vertex cover of a graph (and therefore its Alcuin Number), small boat graphs cannot have a set of forbidden minors. So, instead we are looking for a set of forbidden configurations(to be explained below).

Example 4.1. Consider the graph G below. This graph is small boat as it has 2 vertex covers. $\tau(G) = \{a, d\} = \{a, c\}$.



Here is a subgraph G' of G formed by removing the edge between d and c. This graph is the same as the fox-fox-goat-cabbage graph and is therefore large boat.

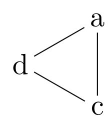
[6] We may assume that Step 5 of the original schedule involves dropping the elements on the right bank, so we can do the same in the new schedule.



Thus, we have demonstrated that removing an edge that changes the vertex cover might make a graph large boat, which means that Alcuin small boat graphs are not in general minor closed.

Definition 4.2. [3, p. 42] *An Induced Subgraph* $G' = (V', E')$ of a graph $G = (V, E)$ is a subgraph where $V' \subset V$ and an edge e' between two vertices v_1, v_2 exists in G' if it also exists in G.

Example 4.3. Consider the graph G from the Example 4.1 above. Here is an induced subgraph H of G.



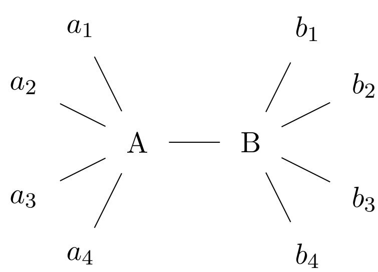
However, the graph H' below is *not* an induced subgraph of G as the edge from a to d exists in G but not in H'.

a — c — d

Definition 4.4. Let $G = (V, E)$ be a graph with minimal vertex cover $\tau(G)$. A *τ-Induced Subgraph* $G' = (V', E')$ is an induced subgraph of G where[7]:

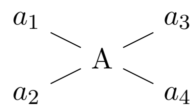
- $\tau(G') \subseteq \tau(G)$
- If a vertex v in $\overline{\tau(G)}$ is adjacent to a vertex in $\tau(G')$, then it is in $\overline{\tau(G')}$.

Example 4.5. Consider the graph G below.

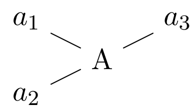


We have the τ-Induced Subgraph G' shown below.

[7]Definitions 4.4 and 4.6 in this section are original.



However, G'' below is not a τ-Induced Subgraph, as a_4 is not included in G'' but it is connected to $A \in \tau(G'')$.



Definition 4.6. Let G' be a graph. G' is *a Small Boat Subgraph (SBS)* if whenever G' is a τ-Induced Subgraph of another graph G, then G is a small boat graph.

Note that if a graph is an SBS, then by the definition above it must also be a small boat graph.

SBSs are at the heart of this thesis as they configurations that inform us about when a graph is small boat, which is what we are looking for! As of now, we are not ready to develop SBSs of our own, so we will need to develop some theory before we can take a look at an example (Example 5.5).

5. Classifying Small Boat Subgraphs

We can categorize the different SBSs G' by the size of their vertex cover $|\tau(G')|$. This allows us to approach the problem for $|\tau(G')| = n$ where $n \in \mathbb{N}$.

First, however, we introduce a useful Lemma and definition that helps us simplify the problem.

5.1. **The maximal number of universal vertices.**

Definition 5.1. *A universal vertex* u of a graph G is a vertex $u \in \overline{\tau(G)}$ such that u is adjacent to every vertex in $\tau(G)$.[8]

Lemma 5.2 (The maximal number of universal vertices [2, Lem 4.4])**.** *Let $G = (V, E)$ have a vertex cover $\tau(G)$ of size n and $\overline{\tau(G)}$ of size m that consists of only universal vertices $u_1, u_2, ...u_m$. Let $\mathbb{A}$ be a maximal Anticlique of $\tau(G)$.*[9]

- *$m \leq 2|\mathbb{A}|$ then G is small boat.*
- *$m > 2|\mathbb{A}|$then G is large boat.*

Proof. Consider a graph $G = (V, E)$ with a vertex cover $\tau(G)$. Say there are n vertices in $\tau(G)$. To prove the first statement, we may assume that

[8] Definitions 5.1, 5.6 and Lemma 5.4 are original work.

[9] I have arrived at this proof independently, but Lemma 4.4 in [2] shows a similar result.

$\overline{\tau(G)}$ consists of $2|\mathbb{A}|$ universal vertices. Let's call these universal vertices $u_1, ..., u_{2|\mathbb{A}|}$.

The largest stable set in $\tau(G)$ is $\mathbb{A}$. We can start the schedule as follows:

$$G \qquad \emptyset \qquad \emptyset \tag{1}$$
$$\overline{\tau(G)} \qquad \tau(G) \qquad \emptyset \tag{2}$$
$$\overline{\tau(G)} \qquad \tau(G) - \mathbb{A} \qquad \mathbb{A} \tag{3}$$

This means there are $|\mathbb{A}|$ spots on the boat now. So, we can now pick up $u_1, ...u_{|\mathbb{A}|}$ in our next step and safely place them on the ending bank while we pick up $\mathbb{A}$ again.

$$\overline{\tau(G)} - \{u_1, ...u_{|\mathbb{A}|}\} \qquad (\tau(G) - \mathbb{A}) \cup \{u_1, ...u_{|\mathbb{A}|}\} \qquad \mathbb{A} \tag{4}$$
$$\overline{\tau(G)} - \{u_1, ...u_{|\mathbb{A}|}\} \qquad \tau(G) \qquad \{u_1, ...u_{|\mathbb{A}|}\} \tag{5}$$

Note that $\overline{\tau(G)} - \{u_1, ...u_{|\mathbb{A}|}\} = \{u_{|\mathbb{A}|+1}, ...u_{2|\mathbb{A}|}\}$. Now, we can replace $u_{|\mathbb{A}|+1}, ...u_{2|\mathbb{A}|}$ with $\mathbb{A}$ on the first bank, and transport the rest of the universal vertices over.

$$\mathbb{A} \qquad \tau(G) - \mathbb{A} \cup \{u_{|\mathbb{A}|+1}, ...u_{2|\mathbb{A}|}\} \qquad \{u_1, ...u_{|\mathbb{A}|}\} \tag{6}$$
$$\mathbb{A} \qquad \tau(G) - \mathbb{A} \qquad \overline{\tau(G)} \tag{7}$$

Finally, it is easy to see the solution.

$$\emptyset \qquad \tau(G) \qquad \overline{\tau(G)} \tag{8}$$
$$\emptyset \qquad \emptyset \qquad G \tag{9}$$

So, if there are at most $2|\mathbb{A}|$ universal vertices that make up $\overline{\tau(G)}$, then G is small boat.

Now all that is left to show is that if there are more than $2|\mathbb{A}|$ universal vertices, we must have a large boat graph. Assume we have $2|\mathbb{A}|+1$ universal vertices.

Suppose for contradiction that the graph with $2|\mathbb{A}|+1$ universal vertices is small boat. Then, this graph has a viable schedule with a boat size of $|\tau(G)|$, and also a viable reverse schedule, which moves elements from the right bank to the left instead of the other way around.

(1) The boat size is $|\tau(G)|$ and the maximum value of $|\mathbb{A}| = |\tau(G)|$. Additionally, we have $2|\mathbb{A}| + 1$ universal vertices, so it is impossible to fit all the universal vertices on the boat at once. Therefore, there must exist some step in the schedule with univeral vertices on both the left and right bank. In this case, the boat must be full with $\tau(G)$ (no empty slots).

(2) Let k be the first such step in the schedule where there are more universal vertices on the right bank than the left. Since there are $2|\mathbb{A}|+1$ universal vertices, at step k, there must be $\geq |\mathbb{A}|+1$ universal vertices on the right bank and $\leq |\mathbb{A}|$ on the left bank. From above, we know that there is at least one universal vertex on both banks at step k.

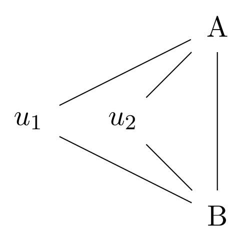
(3) Now, consider the reverse schedule. As stated above, if the graph with $2|\mathbb{A}| + 1$ universal vertices is small boat, it has a viable schedule and a viable reverse schedule, both with boat size $|\tau(G)|$. In this reverse schedule, the step k is now the *last* step with more universals on the right bank.[10]

Therefore, step k is the last step with universals on both banks such that there are more universals on the right bank($\geq |\mathbb{A}| + 1$) than the left bank($\leq |\mathbb{A}|$). We also know that now the boat must hold $\tau(G)$ and that our next step must involve *decreasing* the number of universals on the right bank.

However, we can at most leave $|\mathbb{A}|$ vertices from our boat on the right bank, and there are $\geq |\mathbb{A}| + 1$ universal vertices on the right bank, which means it is impossible for us to make a swap for the remaining universal vertices without conflict. Thus, the reverse schedule from the right bank to the left bank is not viable.

Thus, we have the desired contradiction. □

Example 5.3. Consider the following graph, which has $\tau(G) = \{A, B\}$.

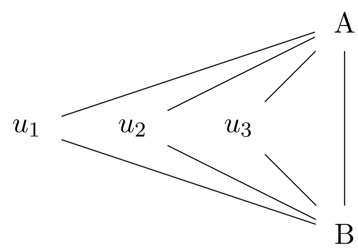


The anticlique number of $\tau(G)$ is 1 (since either $\{A\}$ or $\{B\}$ is the maximal anticlique), and since $2 \leq 2 \cdot 1$, Lemma 5.2 says this graph is small boat. For confirmation, we have the following schedule:

[10]The original schedule transfers elements from the left to the right bank so the reverse transfers from the right bank to the left bank.

(1)	A, B, u_1, u_2	$\emptyset$	$\emptyset$
(2)	u_1, u_2	A, B	$\emptyset$
(3)	u_1, u_2	B	A
(4)	u_2	u_1, B	A
(5)	u_2	A, B	u_1
(6)	A	u_2, B	u_1
(7)	A	B	u_1, u_2
(8)	$\emptyset$	A, B	u_1, u_2
(9)	$\emptyset$	$\emptyset$	A, B, u_1, u_2

Now let's add one more universal vertex, u_3:



The anticlique number of the vertex cover is still 1, and since $3 > 2 \cdot 1$, Lemma 5.2 says this graph is large boat.

The reason Lemma 5.2 is so useful to finding SBSs is that we can start with a graph G with just maximum number of universal vertices ($2|\mathbb{A}|$) in $\overline{\tau(G)}$ and know for a fact that it is small boat. In fact, since the graph is small boat, is is also an SBS. Using G, we can now remove universal vertices and replace them with any other vertex and know that the resulting graph is small boat by the same schedule. So, we can use these graphs with $2|\mathbb{A}|$ universal vertices as a starting point to find more small boat graphs.

Lemma 5.4. *If a graph is small boat, then it is an SBS.*

Proof. Consider any graph G that is small boat. This means it has a viable schedule with boat size $|\tau(G)|$. Now, consider a larger graph H such that G is a τ-Induced Subgraph of H.

We claim that we can solve this H a boat of size $|\tau(H)|$. This ensures that H is a small boat graph and therefore G is an SBS. We first put $\tau(H)$ on the boat, which must include $\tau(G)$ of course. Let $\mathbb{A}_{\mathbb{G}}$ be a maximal anticlique of $\tau(G)$, and $\mathbb{A}_{\mathbb{H}}$ be a maximal anticlique of $\tau(H)$.

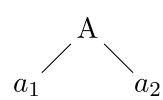
Next, we can leave all the elements of $\tau(H)\backslash\mathbb{A}_{\mathbb{G}}$ on the boat, while we leave all the elements of $\mathbb{A}_{\mathbb{G}}$ on the right bank, and use the $|\mathbb{A}_{\mathbb{G}}|$ empty spots on our boat to shuttle over all the elements in $\overline{\tau(H)}$ that are not connected to $\mathbb{A}_{\mathbb{G}}$. So, during this step, we can certainly shuttle over all the elements in $\overline{\tau(G)} - \overline{\tau(H)}$ as if an element of $\overline{\tau(H)}$ is adjacent to an element of $\tau(G)$, then it must be inside $\overline{\tau(G)}$ as G is a τ-Induced Subgraph of H. In addition to $\overline{\tau(G)} - \overline{\tau(H)}$, we may be able to shuttle over some elements of $\overline{\tau(G)}$ that are not connected to $\mathbb{A}_G$.

Finally, since G is a small boat graph, we can move the the rest of $\overline{\tau(G)}$ over with the $|\mathbb{A}_{\mathbb{G}}|$ slots on the boat by solving the graph G exactly as per its schedule. We know this can always be done using Lemmas 3.5 and 3.7. Since G is a small boat graph, there is a viable schedule that involves us dropping off $\mathbb{A}_{\mathbb{G}}$ on the right bank and then transferring over the rest of G over with the $|\mathbb{A}_{\mathbb{G}}|$ spots opened up on the boat. The other elements $H\backslash G$ have no effect as they do not interact with elements of $\tau(G)$.

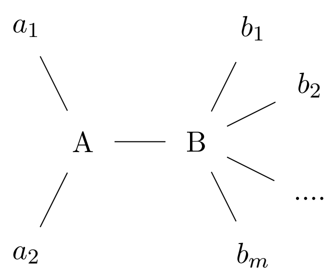
In essence, what we are doing is leaving all the elements of $\tau(H)\backslash\mathbb{A}_{\mathbb{G}}$ on our boat and then solving the G graph with the remaining $|\mathbb{A}_{\mathbb{G}}|$ spots, hence making H a small boat graph. □

Note that since every small boat graph is an SBS, we know that every small boat graph G contains an SBS, which may be improper (i.e. G itself). This implies that a graph is an SBS if and only if it is small boat.

Example 5.5. We've already seen that the fox-goat cabbage graph G is small boat with $|\tau(G)| = 1$[11].



Using Lemma 5.4, G is an SBS. Now consider the following graph H, which has G as a τ-Induced Subgraph (m is an arbitrarily large finite number.)



Then, H is small boat because it contains the SBS G. Here $\tau(H) = \{A, B\}$ and a schedule that illustrates the proof of Lemma 5.4. Note that

[11]We can also confirm this using Lemma 5.4.

in the schedules below, we will express the vertices $\{x_1, ..., x_n\}$ as $\vec{x}$ to avoid clutter and improve readability. This applies to all schedules from this point onwards.

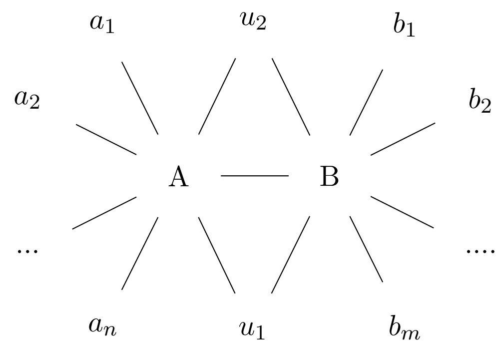
Here is the schedule:

$$
\begin{array}{llll}
(1) & A, B, a_1, a_2, \vec{b} & \emptyset & \emptyset \\
(2) & a_1, a_2, \vec{b} & A, B & \emptyset \\
(3) & a_1, a_2, \vec{b} & B & A \\
(4) & \vdots & \vdots & \vdots \\
(5) & a_1, a_2 & B & A, \vec{b} \\
(6) & a_1 & B, a_2 & A, \vec{b} \\
(7) & a_1 & B, A & a_2, \vec{b} \\
(8) & A & B, a_1 & a_2, \vec{b} \\
(9) & A & B & a_1, a_2, \vec{b} \\
(10) & \emptyset & A, B & a_1, a_2, \vec{b} \\
(11) & \emptyset & \emptyset & A, B, a_1, a_2, \vec{b}
\end{array}
$$

Here, we are keeping B on the boat while following the schedule to show H as a small boat graph.

Definition 5.6. *A maximal set of Alcuin graphs* (referred to as a maximal set.) $\{G\}$ is a set of small boat graphs such that if any vertex v is added to the graph without changing $\tau(G)$, or any edge e is added between two elements of $\tau(G)$, the resulting graph is either still a part of the set or no longer a small boat graph.
It is possible for a maximal set of graphs to be arbitrarily large (See Example 5.8).

Example 5.7. The following set of graphs $\{H\}$ is maximal. This particular set of graphs H includes graphs H with $\tau(H) = \{A, B\}$,2 universal vertices u_1 and u_2, and n vertices a_i connected to A and m vertices b_j connected to B, where n and m are arbitrarily large. All graphs in this set have very similar schedules (See Section 6.13).



We explore the set of graphs in much more detail in Section 6.13, but for now, note that adding another a or b vertex still leaves us in the same set. Therefore, the only vertex/edge we can add to this graph without changing the vertex cover is another universal u_3. However, adding u_3, connected to A and B, however, makes the resulting graph large boat (again, we provide a detailed argument later in Section 6.13). Therefore, the set of graphs of this form is maximal.

In a sense, a maximal set $\{G\}$ is a set of graphs that are as close to being large boat as possible without being large boat. The maximal sets are exactly what we are looking for, as each small boat graph is an induced subgraph of an element of a set and therefore if we find all the maximal sets we can account for all non-maximal SBS graphs underneath them.

Now, we can succinctly state our goal- which is to categorize all the maximal sets for any $|\tau(G)| = n$. In doing so, we also account for all the induced subgraphs (non-maximal SBSs) and therefore account for all the SBSs.

Note that as we categorize the maximal sets of SBSs for $\tau(G) = n$, we will only count graphs that do not contain SBSs with smaller $\tau(G) = m < n$ as these are already accounted for in the catalog for $\tau(G) = m$ SBSs.

5.2. **Naming graphs.** As we attempt to categorize SBSs for larger $|\tau(G)|$, the graphs become increasingly complicated. This section aims to minimize clutter and improve efficiency by assigning a naming convention[12] by which each SBS can be named based on unique characteristics. The naming convention goes as follows:

For a graph with $|\tau(G)| = n$, we assign the name:

$$G_{n,\mathbb{A},v_1,v_2,v_3,\ldots,v_{n-1};u_n}$$

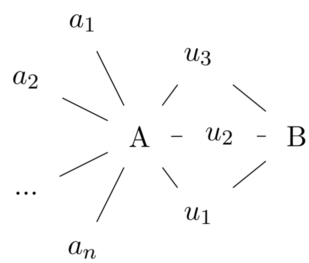
where v_1 is the number of elements of $\tau(G)$ with linked degree-one vertices in $\overline{\tau(G)}$, v_2 is the number of unordered pairs of elements of $\tau(G)$ each connected to degree-two vertices in $\overline{\tau(G)}$, and v_i is the number of unordered i-tuples

[12] This naming convention is original work, so there are no references attached.

in $\tau(G)$ connected to degree-i vertices in $\overline{\tau(G)}$. Finally u_n is the number of universal vertices. If $v_1, v_2, ..., v_{n-1}$ terminates in a string of zeros, then we cut out those zeros.

Let's look at a few examples.

Example 5.8. The graph below has $|\tau(G)| = 2$, and $|\mathbb{A}| = 2$. It also has degree one vertices $\{a_1, ..., a_n\}$ only connected to one element of $\tau(G)$, and finally, 3 universal vertices $\{u_1, u_2, u_3\}$. Therefore, we call it $G_{2,2,1;3}$.



Example 5.9. The fox-goat-cabbage graph has $|\tau(G)| = 1$, and $|\mathbb{A}| = 1$. Additionally, it has 2 universal vertices $\{u_1, u_2\}$. Therefore, we call it $G_{1,1;2}$.

$$a_1 — \mathrm{A} — a_2$$

Going forward, we will use this naming convention for all SBSs.

5.3. **The case of $|\tau(G)| = 1$.**

Example 5.10. Continuing Example 5.9, consider the fox-goat-cabbage graph $G_{1,1;2}$. Consider the set $\{G_{1,1;2}\}$. This set only contains a singular element. [13]

$$u_1 — \mathrm{A} — u_2$$

In this case, $\tau(G) = \{A\}$. The schedule is:

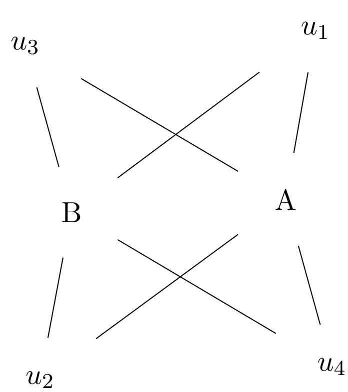
[13]The classification of graphs in Sections 5.3 and 5.4 is original work. There are no references.

(1)	A, a, a	$\emptyset$	$\emptyset$
(2)	a, a	A	$\emptyset$
(3)	a, a	$\emptyset$	A
(4)	a	a	A
(5)	a	A	a
(6)	A	a	a
(7)	A	$\emptyset$	a, a
(8)	$\emptyset$	A	a, a
(9)	$\emptyset$	$\emptyset$	A, a, a

By Lemma 5.2, we cannot add any vertices connected to A without making $G_{1,1;2}$ graph large boat. Hence, the set $\{G_{1,1;2}\}$ is maximal. Removing any of the universal vertices is redundant as the resulting graph (although small boat) would have the same schedule as $G_{1,1;2}$ and therefore is already accounted for in as a subgraph of $G_{1,1;2}$ in the maximal set $\{G_{1,1;2}\}$. So, $\{G_{1,1;2}\}$ is the only maximal SBS with $|\tau(G)| = 1$.

5.4. **The case of $|\tau(G)| = 2$.**

Example 5.11. As we move to higher $|\tau(G)|$, finding all the SBSs becomes much more complicated. We can start with the maximum number of universal vertices for a graph with $|\tau(G)| = 2$, which is the set $\{G_{2,2,0;4}\}$ (represented below). This set also contains only one element for the same reasons as $\{G_{1,1;2}\}$.



In this case, $\tau(G) = \{A, B\}$. The schedule is:

(1)	A, B, u_1, u_2, u_3, u_4	$\emptyset$	$\emptyset$
(2)	u_1, u_2, u_3, u_4	A, B	$\emptyset$
(3)	u_1, u_2, u_3, u_4	$\emptyset$	A, B
(4)	u_3, u_4	u_1, u_2	A, B
(5)	u_3, u_4	A, B	u_1, u_2
(6)	A, B	u_3, u_4	u_1, u_2
(7)	A, B	$\emptyset$	u_1, u_2, u_3, u_4
(8)	$\emptyset$	A, B	u_1, u_2, u_3, u_4
(9)	$\emptyset$	$\emptyset$	A, B, u_1, u_2, u_3, u_4

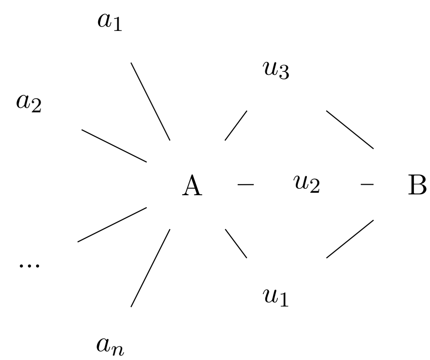
This graph is small boat by Lemma 5.2 and a SBS by Lemma 5.4 respectively. If we add a single universal vertex, the graph is large boat by Lemma 5.4. If we add a degree-one vertex connected to A or B, it is easy to check that the resulting graph is also large boat. [14] Additionally, connecting A to B reduces the anticlique number $\mathbb{A}$ and also makes this graph large boat by Lemma 5.2. So, this graph set is maximal.

Now that we have a first example of a maximal set, we can reduce the number of universal vertices and add other edges in their place to create other maximal small boat subgraphs. So, Lemma 5.4 gives us a "starting point", i.e. a small boat graph with a maximal number of universal vertices from which we can generate other SBSs with $|\tau(G)| = 2$.

Example 5.12. Let's prune one universal vertex u_4. Now, we have some "space" to add more vertices to $\tau(G)$. We also ensure that our graph set has the most amount of vertices possible, giving us a maximal set with 3 universal vertices.

We can add degree-one vertices connected to A or B. One can check that if we add one degree-one vertex to each of A and B, our graph becomes large boat. So, without loss of generality, we can add degree-one vertices to A. We claim that we can add an arbitrary number (n) of degree-one vertices. This gives the raph set $\{G_{2,2,1;3}\}$ below. This set has an arbitrarily large number of graphs, similar to the one in Example 6.8.

[14] Note that any vertex added must connect to $\tau(G)$ in order to not change the vertex cover as per Definition 5.6.

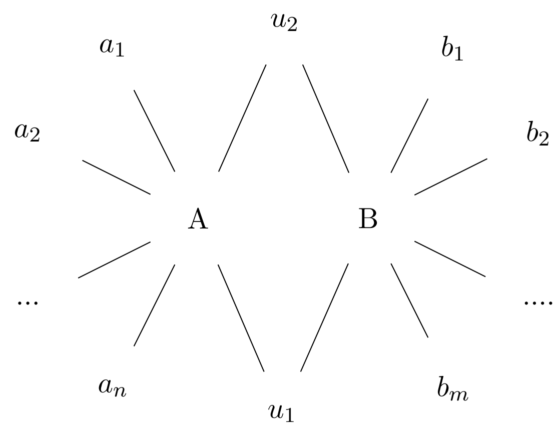


In this case, $\tau(G) = \{A, B\}$. The schedule is:

(1)	$A, B, u_1, u_2, u_3, \vec{a}$	$\emptyset$	$\emptyset$
(2)	$u_1, u_2, u_3, \vec{a}$	A, B	$\emptyset$
(3)	$u_1, u_2, u_3, \vec{a}$	A	B
(4)	$\vdots$	$\vdots$	$\vdots$
(5)	u_2, u_3	A, u_1	$B, \vec{a}$
(6)	u_2, u_3	A, B	$u_1, \vec{a}$
(7)	A, B	u_2, u_3	$u_1, \vec{a}$
(8)	A, B	$\emptyset$	$u_1, u_2, u_3, \vec{a}$
(9)	$\emptyset$	A, B	$u_1, u_2, u_3, \vec{a}$
(10)	$\emptyset$	$\emptyset$	$A, B, u_1, u_2, u_3, \vec{a}$

At this point, we have shown that this graph set is small boat and therefore a SBS. We cannot add vertices connected to A, as in that case we are still considering the same graph set $\{G_{2,2,1;3}\}$, and one can check that adding a single degree-one vertex to B or a universal vertex to A and B makes this graph set large boat. Finally, adding the edge between A and B also makes the graph set large boat. So, we can no longer add vertices or edges without changing $|\tau(G)|$, which means that the set above is maximal.

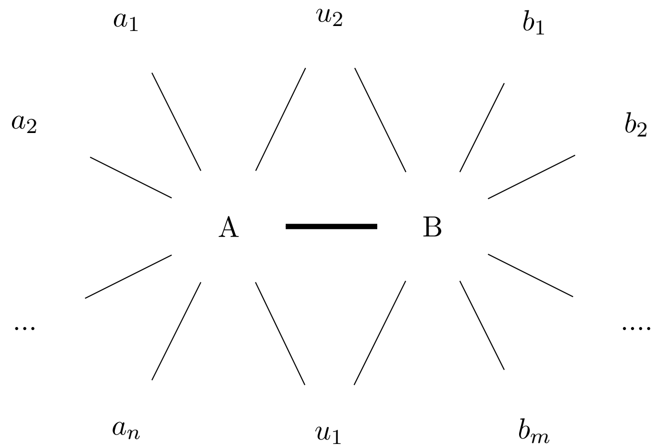
Example 5.13. Next, prune one more universal vertex u_3. Now, we can add an arbitrary number of degree-one vertices to A and B like so. (m and n are arbitrarily large finite numbers.) This gives the graph set $\{G_{2,2,2;2}\}$ below, with an arbitrarily large number of graphs.



The vertex cover here is $\tau(G) = \{A, B\}$. The schedule is:

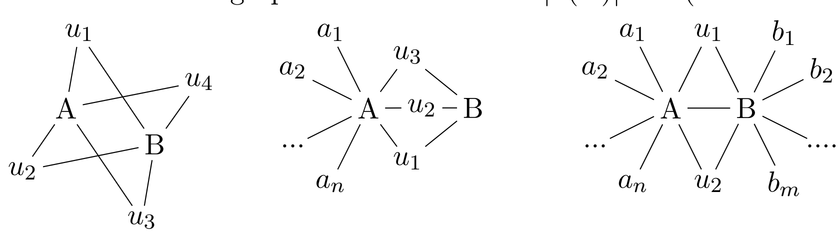
(1)	$A, B, u_1, u_2, \vec{a}, \vec{b}$	$\emptyset$	$\emptyset$
(2)	$u_1, u_2, \vec{a}, \vec{b}$	A, B	$\emptyset$
(3)	$u_1, u_2, \vec{a}, \vec{b}$	A	B
(4)	$\vdots$	$\vdots$	$\vdots$
(5)	$u_2, \vec{b}$	A, u_1	$B, \vec{a}$
(6)	$u_2, \vec{b}$	A, B	$u_1, \vec{a}$
(7)	$A, \vec{b}$	B, u_2	$u_1, \vec{a}$
(8)	$A, \vec{b}$	B	$u_1, u_2, \vec{a}$
(9)	$\vdots$	$\vdots$	$\vdots$
(8)	A	B	$u_1, u_2, \vec{a}, \vec{b}$
(9)	$\emptyset$	A, B	$u_1, u_2, \vec{a}, \vec{b}$
(10)	$\emptyset$	$\emptyset$	$A, B, u_1, u_2, \vec{a}, \vec{b}$

Example 5.14. Continuing with Example 5.13, we can no longer add more degree-one vertices to A or B as we are still in the same graph set $\{G_{2,2,2;2}\}$. One can check that we cannot add another universal vertex either. But, we can add the edge between A and B. This new set is $\{G_{2,1,2;2}\}$ (drawn below).



It turns out that this graph set ($G_{2,1,2;2}$) is also small boat, and by the exact same schedule as graph set $\{G_{2,2,2;2}\}$ (Example 6.13). So, the previous set was not maximal, and this one is now maximal[15]. This is because we cannot add any single degree vertices a or b as that still leaves us in the same graph set $\{G_{2,1,2;2}\}$. Additionally, we cannot add any edges without changing $|\tau(G)|$. Finally, one can check that adding a universal vertex makes the resulting graph set large boat. Therefore, this the graph set $\{G_{2,1,2;2}\}$ is maximal.

At this point, any degree one vertices connected to A or B result in a graph in the same maximal set. Additionally, we can not add any edges without changing $|\tau(G)|$. So, there are no more vertices or edges left to add if we remove one universal vertex, which is why we can now claim that we have found all the maximal graph sets of SBSs with $|\tau(G)| = 2$ (Listed below).



Therefore, all small boat subgraphs with $|\tau(G)| = 2$ must be induced subgraphs of graphs from the three(maximal) graph sets listed above.

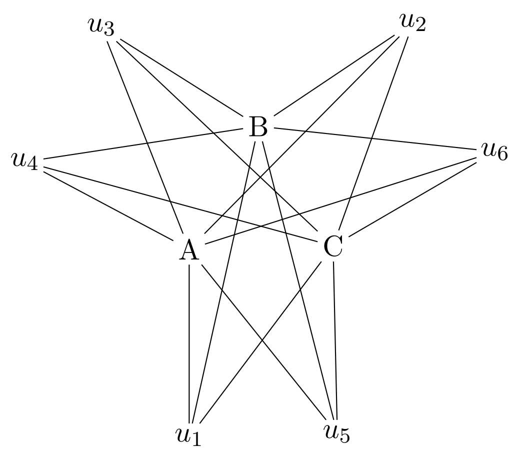
5.5. **The case of** $|\tau(G)| = 3$**.** After going through a similar process, we can isolate all the maximal sets of SBSs with $|\tau(G)| = 3$. (The examples were exhaustively checked but the details are not provided here).[16]

First, using Lemma 5.2, we have $\{G_{3,3,0,0;6}\}$ below.

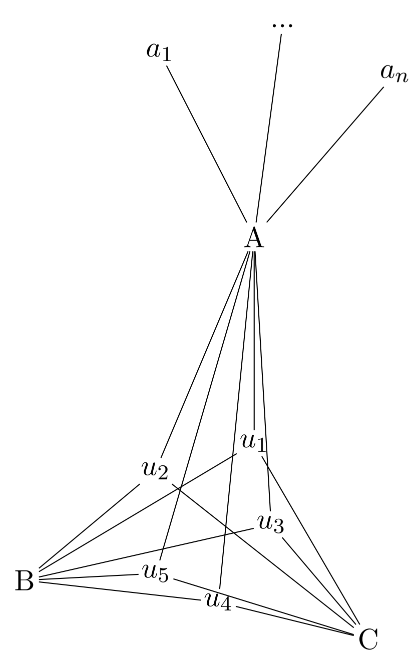
---

[15]Note that this is the graph from Example 3.8

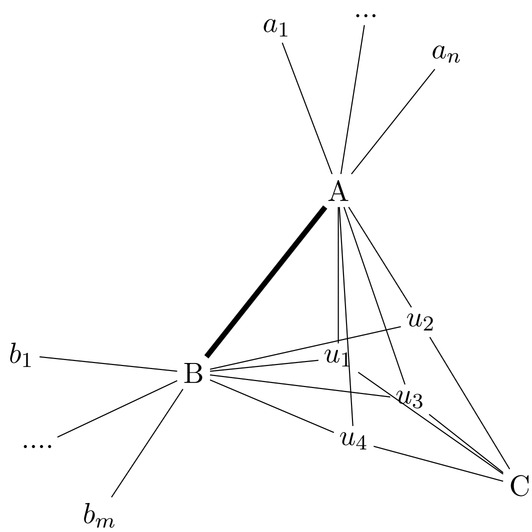
[16]The same applies for the unjustified claims in sections for $|\tau(G)| = 1$, $|\tau(G)| = 2$. These classifications are original work and so there are no references attached.



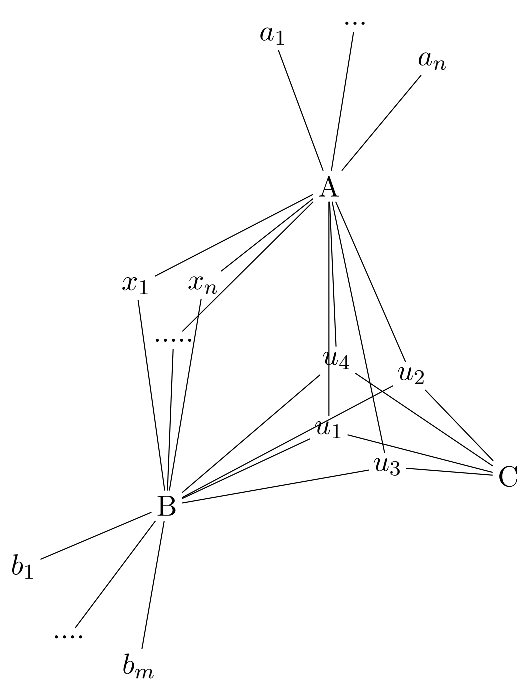
Reducing the number of universal vertices by 1, we arrive at $\{G_{3,3,1,0;5}\}$.



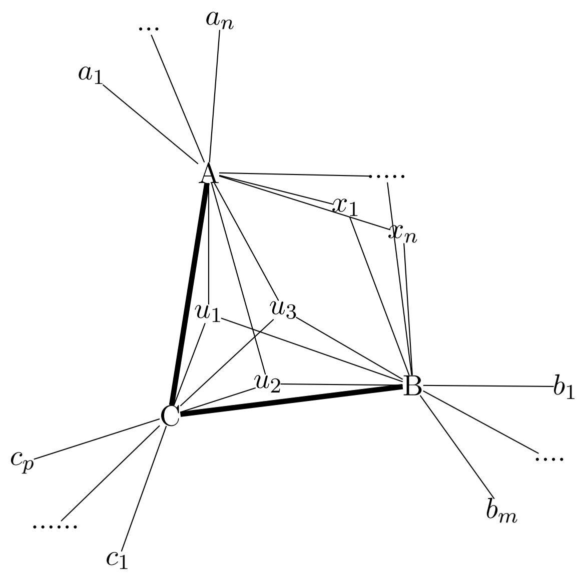
From here, when we drop one universal, we are able to achieve 2 different graphs with 4 universal vertices each. The first one, presented below is $\{G_{3,2,2,0;4}\}$.



We also have $\{G_{3,3,2,1;4}\}$.



Dropping a universal from $\{G_{3,2,2,0;4}\}$ creates a graph containing $\{G_{2,1,2;2}\}$ as an SBS, which means it is not counted in the list of SBSs with $|\tau(G)| = 3$. However, dropping a universal vertex from $\{G_{3,3,2,1;4}\}$ gives our final SBS: $\{G_{3,2,3,1;3}\}$ as shown below.



All other SBSs are not maximal or contain SBSs with smaller $|\tau(G)|$ within them and therefore are not counted. These 5 graph sets above ($\{G_{3,3,1,0;5}\}$, $\{G_{3,3,1,0;5}\}$, $\{G_{3,2,2,0;4}\}$, $\{G_{3,3,2,1;4}\}$,$\{G_{3,2,3,1;3}\}$) then are the complete list of maximal sets of SBSs with $|\tau(G)| = 3$.

6. Splitting Boats

As we move to higher $|\tau(G)|$, SBSs become increasingly complicated and hard to check. Our method for manually finding all maximal sets of SBSs becomes inefficient, and we require other methods to find the maximal sets of SBSs. This section aims at building maximal SBSs of larger $|\tau(G)|$ from smaller SBSs. [17]

Definition 6.1. We can *add* two SBSs G_1 and G_2 together (to form a new graph G_3) with one simple step. We take the disjoint union and then connect the universal vertices of G_1 to $\tau(G_2)$ and the universal vertices of G_2 to $\tau(G_1)$, thereby maintaining their universality [18].

Lemma 6.2. *If we* add *together two SBSs G_1 and G_2 to form a graph G_3, then the resulting G_3 is also an SBS.*

Proof. If we add SBSs G_1 and G_2 to form G_3, then to show that G_3 is also an SBS, we need to prove that we can solve G_3 with a boat size of $|\tau(G_3)|$. We know that since G_1 and G_2 are each SBSs, they each have a viable schedule with boat size $|\tau(G_1)|$ (say S_1) and $|\tau(G_2)|$ (say S_2) respectively

[17] Section 6 is original work, so there are no references.

[18] So they remain connected to all vertices in $\tau(G_3)$

(These schedules might not be of the same length). We also know that $|\tau(G_3)| = |\tau(G_1)| + |\tau(G_2)|$, since when we add G_1 and G_2, neither vertex cover changes, and we simply link the universal vertices to the other vertex cover.

So, our boat must have a size of $|\tau(G_1)| + |\tau(G_2)|$, which means we can "split" it into two segments B_1 and B_2. B_1 will be the first $|\tau(G_1)|$ slots on the boat, and will be assigned to G_1, and similarly B_2 will be the next $|\tau(G_2)|$ slots assigned to G_2. With this in mind, let's begin.

6.0.1. *Step 1:* First, we must transport the vertex cover $\tau(G_3) = \tau(G_1) + \tau(G_2)$. Following the individual schedules for the SBSs G_1 and G_2, we first pick up $\tau(G_1)$ (on B_1) and $\tau(G_2)$ (on B_2).

6.0.2. *Step 2:* Again, following the individual schedules (S_1 and S_2), we can now leave $\mathbb{A}_1$ and $\mathbb{A}_2$ on the right bank (as per Lemma 3.7)[19]. At this point, B_1 has $|\mathbb{A}_1|$ empty slots and B_2 has $|\mathbb{A}_2|$ empty slots.

6.0.3. *Step 3:* Now, we must continue to follow the schedules S_1 and S_2. We use the $|\mathbb{A}_1|$ and $|\mathbb{A}_2|$ open slots on B_1 and B_2 to transfer over any vertices that do not interact with any of the elements of $\mathbb{A}_1$ and $\mathbb{A}_2$ to the right bank. We can transfer over all of these without moving anything off the right bank, so we can choose our schedules S_1 and S_2 so that they move all these vertices over without leaving any on the left bank. Since these vertices do not interact with $\mathbb{A}_1$ and $\mathbb{A}_2$, we do not have to remove anything from the right bank yet.

Note that at this step, the schedules might differ in length. One schedule might have more such vertices to transfer over, and in this case, the other segment must wait until both segments are finished with this process.

At this point, we should have transferred over all the vertices that do not interact with $\mathbb{A}_1$ and $\mathbb{A}_2$, and still have $|\mathbb{A}_1|$ and $|\mathbb{A}_2|$ slots open on B_1 and B_2 respectively.

6.0.4. *Step 4:* Now, any material that we transfer over will require that we move elements of $\mathbb{A}_1$ and $\mathbb{A}_2$ back onto B_1 and B_2. Note that since the universal vertices of G_1 are now linked to $\tau(G_2)$ and the universal vertices of G_2 are linked to $\tau(G_1)$, if we move a single universal vertex over, both $\mathbb{A}_1$ and $\mathbb{A}_2$ will need to be moved back onto B_1 and B_2, filling them up completely [20].

So, at this point, we must choose to move over at most $|\mathbb{A}_1|$ and $|\mathbb{A}_2|$ elements over and we can break the problem down into cases.

[19] Since $\mathbb{A}_1$ and $\mathbb{A}_2$ are the maximal anticliques for G_1 and G_2, $\mathbb{A}_1 + \mathbb{A}_2 = \mathbb{A}_3$ becomes the maximal anticlique for the sum G_3.

[20] Recall that they have $|\mathbb{A}_1|$ and $|\mathbb{A}_2|$ spots open respectively

Either both segments B_1 and B_2 contain a universal vertex (Case 1), only one segment B_1 or B_2 contains a universal vertex (Case 2) or neither segment contains a universal vertex (Case 3).

- *In Case 1*- Both segments contain a universal vertex. In this case, we can proceed as per the schedules S_1 and S_2. We will transfer over the $|\mathbb{A}_1|$ and $|\mathbb{A}_2|$ vertices as per S_1 and S_2 respectively, and then move $\mathbb{A}_1$ and $\mathbb{A}_2$ back onto the segments B_1 and B_2 leaving us with B_1 holding $\tau(G_1)$ and B_2 holding $\tau(G_2)$.

At this point, we must swap out $|\mathbb{A}_1|$ and $|\mathbb{A}_2|$ for the remaining elements on the left bank as per S_1 and S_2. There will be no universal vertices left on the left bank after this because if there were, we wouldn't be to swap any element from the boat B_1 or B_2 and this would make S_1 or S_2 unviable schedules.

At this point, all we have to do is shuttle the $|\mathbb{A}_1|$ and $|\mathbb{A}_2|$ elements we just picked up across to the right bank, and return one last time for $\mathbb{A}_1$ and $\mathbb{A}_2$, to empty the left bank, leaving us with $\tau(G_1)$ and $\tau(G_2)$ on the boat. Finally, we transfer $\tau(G_1)$ and $\tau(G_2)$ over, and the schedule is complete[21].

- *In Case 2*- One segment contains a universal vertex and the other does not. Without loss of generality, let's say that the next step for S_1 involves transferring at least one universal vertex to the right bank, and therefore moving $\mathbb{A}_1$ onto B_1, and the next step for S_2 does not involve a universal vertex, rather some other vertices that require us to move only a part of $\mathbb{A}_2$, lets say $Q \subset \mathbb{A}_2$ onto the boat, leaving us with some space.

In this case, we can "pause" the progress of S_1 while we transfer the other vertices using B_2 to the right bank and transfer Q onto B_2. As we follow the schedule S_2, we might have to transfer more and more vertices into Q until one of two things happens.

The first is that $Q = \mathbb{A}_2$ and B_2 is filled.

The second is that Q remains a proper subset of $\mathbb{A}_2$ and the next step in the schedule S_2 is to transfer over a universal vertex and thus move the rest of $\mathbb{A}_2$ back onto B_2. [22]

In both cases, the next step in S_2 involves us filling up B_2 with $\mathbb{A}_2$, and so we are able to "unpause" S_1 and transfer the universal vertex in B_1 over now. At this point, all of $\mathbb{A}_1$ will also need to be moved onto B_1 as there is now a universal vertex on the right bank, as per schedule S_1.

[21]We can always transfer over the remaining vertices with the $|\mathbb{A}_1|$ and $|\mathbb{A}_2|$ slots opened up by dropping $\mathbb{A}_1$ and $\mathbb{A}_2$ on the left bank because in Step 3 we transferred all vertices that do not interact with $\mathbb{A}_1$ and $\mathbb{A}_2$ over already!

[22]It is technically possible to end up with a graph where Q is not equal to $\mathbb{A}_2$ AND there are no universal vertices left to transfer, but this means that there are no vertices in $\overline{\tau(G_2)}$ attached to one element of $\tau(G_2)$, which immediately makes the resulting G_3 small boat as it then contains an induced subgraph of the maximal SBS $G_1, 1, 2$ (Fox-Goat-Cabbage Graph).

So, now both of our schedules S_1 and S_2 have reached a point where B_1 holds $\tau(G_1)$ and B_2 holds $\tau(G_2)$ with no extra room on the boat. Now, we are in Case 1, as all that is left to do is swap $\mathbb{A}_1$ and $\mathbb{A}_2$ for the last $|\mathbb{A}_1|$ and $|\mathbb{A}_2|$ elements on the left bank as described above (in Case 1).

We transfer over the last $|\mathbb{A}_1|$ and $|\mathbb{A}_2|$ elements, return for $\mathbb{A}_1$ and $\mathbb{A}_2$ and transfer over $\tau(G_1)$ and $\tau(G_2)$ to finish the schedule successfully.

• *In Case 3*- Neither segment contains a universal vertex. In this case, we again can proceed as per the schedules S_1 and S_2. We can transfer over $|\mathbb{A}_1|$ and $|\mathbb{A}_2|$ vertices as per S_1 and S_2 respectively, and then move the relevant clashing subsets of $\mathbb{A}_1$ and $\mathbb{A}_2$ (say P and Q) back onto the segments B_1 and B_2 leaving us with B_1 holding $\tau(G_1)-\mathbb{A}_1+P$ and B_2 holding $\tau(G_2)-\mathbb{A}_2+Q$, where $P\subset\mathbb{A}_1$ and $Q\subset\mathbb{A}_2$.

Following this, we continue to shuttle over vertices that clash with the remaining parts of $\mathbb{A}_1$ and $\mathbb{A}_2$ on the right bank, and move the clashing vertices of $\mathbb{A}_1$ and $\mathbb{A}_2$ onto the boat to join P and Q.

(1) We may continue in this fashion (following S_1 and S_2) until $P=\mathbb{A}_1$ and $Q=\mathbb{A}_2$ which makes both boats full. Note that one of P or Q might become equal to $\mathbb{A}_1$ or $\mathbb{A}_2$ respectively before the other, and in this case, we must wait for the other segment to catch up.
After we are at the point when $P=\mathbb{A}_1$ and $Q=\mathbb{A}_2$, all we have to do is continue with the schedules S_1 and S_2 as at this point, both segments have no space left on them. B_1 contains $\tau(G_1)$ and B_2 contains $\tau(G_2)$ and we must return to the left bank.
If there are any universal vertices in G_1, then $\mathbb{A}_1$ will be moved to the left bank and they will all be moved onto the boat. Similarly, all universals from G_2 will be moved onto the boat, leaving $\mathbb{A}_1$ and $\mathbb{A}_2$ on the left bank with no other clashing vertices. We can now shuttle over all the universals to the right bank, and return with $|\mathbb{A}_1|$ and $|\mathbb{A}_2|$ spots on each segment respectively.
From here, the universal vertices are already transferred over, so all we have to do is continue to follow the steps of S_1 and S_2 and we should be able to transfer the remaining vertices over with the $|\mathbb{A}_1|$ and $|\mathbb{A}_2|$ spots on our boat before finally returning one last time for $\mathbb{A}_1$ and $\mathbb{A}_2$. Then, the left bank is empty, and our boat contains $\tau(G_1)$ and $\tau(G_2)$, which we take to the right side and complete the problem.

(2) Another possibility, is that as we move vertices over, one segment, say B_1, contains a universal vertex that clashes with the remaining elements of $\mathbb{A}_2$. In this case, we follow the steps of Case 2 and are able to solve the problem easily.

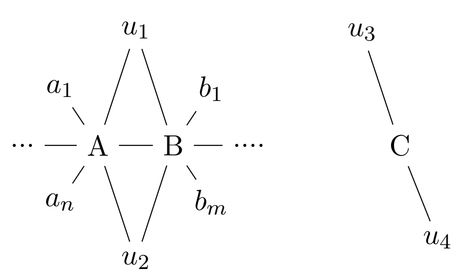
(3) Finally, the last possibility is that both segments B_1 and B_2 contain a universal vertex that clashes with the remaining elements of $\mathbb{A}_2$ and $\mathbb{A}_1$ respectively. In this case, we simply follow the steps of Case 1 and are able to solve the problem easily.

So, we have shown that if we *add* two small boat graphs (and therefore SBSs) G_1 and G_2, then the resulting G_3 is also a small boat graph (and an SBS).

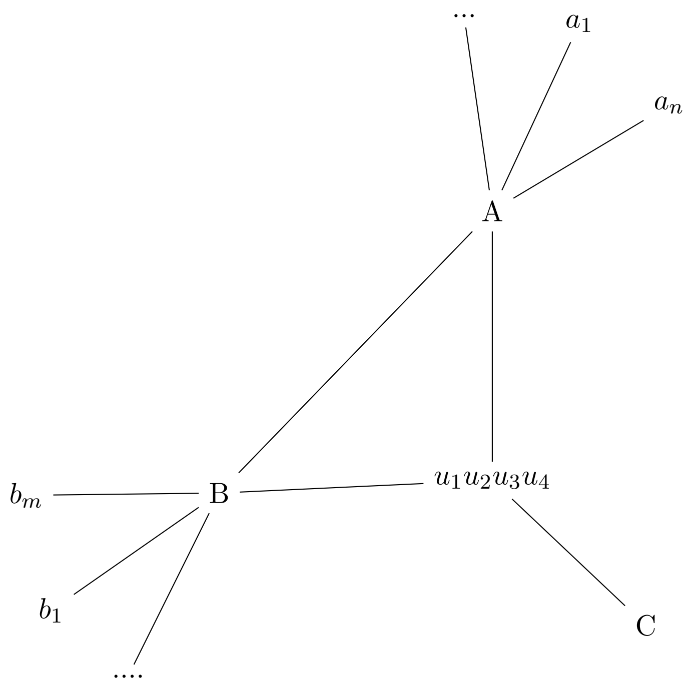
Note that since the universal vertices of G_2 are linked to $\tau(G_1)$ and the universal vertices of G_1 are linked to $\tau(G_2)$ the new graph G_3 is a unique configuration that is not an SBS under G_1 or G_2, rather a unique small boat graph itself. □

Note that this proof is a method for forming unique SBS graphs, but does not necessarily mean that all SBS graphs can be formed via this method from smaller SBS graphs. Finally, note that the SBS sets that are formed via this method might not be maximal (see Example 6.4). Despite these drawbacks, this method allows us to find new SBSs with ease, after which we can also easily check if the result is maximal before adding it to our list for any $|\tau(G)| = n$.

Example 6.3. We can add together the graphs $G_{2,1,2;2}$ and $G_{1,1;2}$ to create the graph $G_{3,2,2,0;4}$. Here are $G_{2,1,2;2}$ and $G_{1,1;2}$ below:



When added together, we get $G_{3,2,2,0;4}$, with 4 universal vertices, $\{a_1, ..., a_n\}$, $\{b_1, ..., b_m\}$ and an edge between A and B.



Here is the schedule as per Lemma 6.2. We start with the vertex covers $\{A, B\}$ and $\{C\}$. Then drop off $\mathbb{A}_1$ and $\mathbb{A}_2$

(1)	$A, B, C, u_1, u_2, u_3, u_4, \vec{a}, \vec{b}$	$\emptyset$	$\emptyset$
(2)	$u_1, u_2, u_3, u_4, \vec{a}, \vec{b}$	A, B, C	$\emptyset$
(3)	$u_1, u_2, u_3, u_4, \vec{a}, \vec{b}$	A	B, C

Now, since B_2 has no vertices it can bring over in Step 4, we can wait for B_1 to transfer over $\{a_1, ..., a_n\}$. Then, we bring over the first two universals (we are in Case 1) one from S_1 and one from S_2.

(4)	$\vdots$	$\vdots$	$\vdots$
(5)	$u_3, u_4, \vec{b}$	A, u_1, u_2	$B, C, \vec{a}$
(6)	$u_3, u_4, \vec{b}$	A, B, C	$u_1, u_2, \vec{a}$

Now, we continue with S_1 and S_2 as we swap $\mathbb{A}_1$ and $\mathbb{A}_2$ onto the left bank as we shuttle over the last of the universal vertices $\{u_3, u_4\}$ (as described in Case 1 in Lemma 6.2).

(7)	$A, C, \vec{b}$	B, u_3, u_4	$u_1, u_2, \vec{a}$
(8)	$A, C, \vec{b}$	B	$u_1, u_2, u_3, u_4, \vec{a}$

At this point, we can use the remaining slot of B_1 to shuttle over the remaining vertices on the left bank $\{b_1, ..., b_m\}$, after which we return one last time for the vertex cover.

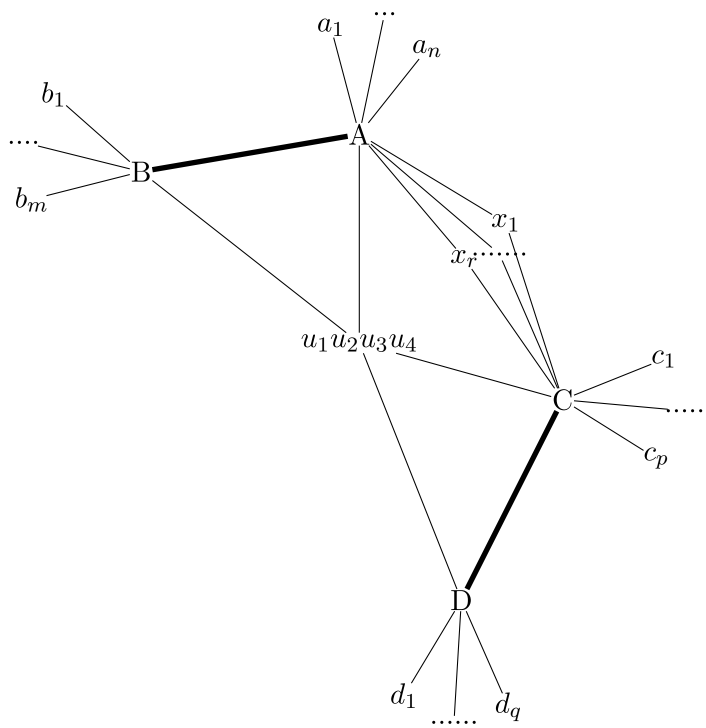
(9)	$\vdots$	$\vdots$	$\vdots$
(8)	A, C	B	$u_1, u_2, u_3, u_4, \vec{a}, \vec{b}$
(9)	$\emptyset$	A, B, C	$u_1, u_2, u_3, u_4, \vec{a}, \vec{b}$
(10)	$\emptyset$	$\emptyset$	$A, B, C, u_1, u_2, u_3, u_4, \vec{a}, \vec{b}$

It turns out that $G_{3,2,2,0;4}$ is maximal as we checked for it in Section 6.5.In fact, 3 of the 5 maximal sets of SBSs with $|\tau(G)| = 3$ can be created using this addition technique (all being maximal).

- $G_{1,1;2} + G_{2,2,0;4} = G_{3,3,0,0;6}$
- $G_{1,1;2} + G_{2,2,1;3} = G_{3,3,1,0;5}$
- $G_{1,1;2} + G_{2,1,2;2} = G_{3,2,2,0;4}$ (Described above)

Example 6.4. It is possible, as stated above, to add two maximal SBS graphs for a resulting graph that is not maximal. Consider the addition $G_{2,1,2;2} + G_{2,1,2;2} = G_{4,2,4,0,0;4}$.

The graph $G_{4,2,4,0,0;4}$ is not maximal as we can add degree two vertices to create the small boat graph $G_{4,2,4,1,0;4}$ (Shown below).



Here is the attempted schedule for $G_{4,2,4,1,0;4}$ that follows the proof of Lemma 6.2:

(1)	$A, B, C, D, u_1, u_2, u_3, u_4, \vec{a}, \vec{b}, \vec{c}, \vec{d}, \vec{x}$	$\emptyset$	$\emptyset$
(2)	$u_1, u_2, u_3, u_4, \vec{a}, \vec{b}, \vec{c}, \vec{d}, \vec{x}$	A, B, C, D	$\emptyset$
(3)	$u_1, u_2, u_3, u_4, \vec{a}, \vec{b}, \vec{c}, \vec{d}, \vec{x}$	A, C	B, D

Now we can shuttle over $\{a_1, ..., a_n\}, \{b_1, ..., b_m\}, \{x_1, ..., x_r\}$, and then move over the first universal vertices u_1 and u_2.

(4)	$\vdots$	$\vdots$	$\vdots$
(5)	$u_1, u_2, u_3, u_4, \vec{b}, \vec{d}$	A, C	$B, D, \vec{a}, \vec{c}, \vec{x}$
(6)	$u_3, u_4, \vec{b}, \vec{d}$	A, C, u_1, u_2	$B, D, \vec{a}, \vec{c}, \vec{x}$
(7)	$u_3, u_4, \vec{b}, \vec{d}$	A, C, B, D	$u_1, u_2, \vec{a}, \vec{c}, \vec{x}$

Finally, we swap A, C for u_3 and u_4 and finish the schedule.

(8)	$A, C, \vec{b}, \vec{d}$	u_3, u_4, B, D	$u_1, u_2, \vec{a}, \vec{c}, \vec{x}$
(9)	$A, C, \vec{b}, \vec{d}$	B, D	$u_1, u_2, u_3, u_4, \vec{a}, \vec{c}, \vec{x}$
(10)	A, C	$B, D, \vec{b}, \vec{d}$	$u_1, u_2, u_3, u_4, \vec{a}, \vec{c}, \vec{x}$
(11)	A, C	B, D	$u_1, u_2, u_3, u_4, \vec{a}, \vec{c}, \vec{x}, \vec{b}, \vec{d}$
(12)	$\emptyset$	A, C, B, D	$u_1, u_2, u_3, u_4, \vec{a}, \vec{c}, \vec{x}, \vec{b}, \vec{d}$
(13)	$\emptyset$	$\emptyset$	$A, B, C, D, u_1, u_2, u_3, u_4, \vec{a}, \vec{c}, \vec{x}, \vec{b}, \vec{d}$

So, the graph $G_{2,1,2;2} + G_{2,1,2;2} = G_{4,2,4,0,0;4}$ is not maximal as we have shown the graph $G_{4,2,4,1,0;4}$, that is formed by adding degree two vertices $\{x_1, ..., x_r\}$ to $G_{4,2,4,0,0;4}$ is a small boat graph.

This serves as a counterexample to the fact that the sum of two maximal small boat graphs is also maximal.

References

[1] R. Balakrishnan and K. Ranganathan. *A Textbook of Graph Theory*. en. Universitext. New York, NY: Springer, 2012. DOI: 10.1007/978-1-4614-4529-6.

[2] Péter Csorba, Cor A. J. Hurkens, and Gerhard J. Woeginger. “The Alcuin Number of a Graph and Its Connections to the Vertex Cover Number”. In: *SIAM Review* 54.1 (Jan. 2012), pp. 141–154. DOI: 10.1137/110848840.

[3] Darij Grinberg. *An introduction to graph theory*. arXiv:2308.04512 [math]. Aug. 2023. DOI: 10.48550/arXiv.2308.04512.

[4] Neil Robertson and P. D. Seymour. “Graph minors. I. Excluding a forest”. In: *Journal of Combinatorial Theory, Series B* 35.1 (Aug. 1983), pp. 39–61. DOI: 10.1016/0095-8956(83)90079-5.

[5] Erfang Shan and Liying Kang. “The Ferry Cover Problem on Regular Graphs and Small-Degree Graphs”. en. In: *Chinese Annals of Mathematics, Series B* 39.6 (Nov. 2018), pp. 933–946. DOI: 10.1007/s11401-018-0105-5.