Theory and Applications of Graphs

Volume 5 | Issue 1 Article 2

January 2018

Traveling in Networks with Blinking Nodes

Braxton Carrigan
Southern CT State University, carriganb1@southernct.edu

James Hammer
Cedar Crest College, jmhammer@cedarcrest.edu

Recommended Citation

Carrigan, Braxton and Hammer, James (2018) "Traveling in Networks with Blinking Nodes," *Theory and Applications of Graphs*: Vol. 5 : Iss. 1 , Article 2.
DOI: 10.20429/tag.2018.050102
Available at: https://digitalcommons.georgiasouthern.edu/tag/vol5/iss1/2

Abstract

We say that a blinking node system modulo n is an ordered pair (G, L) where G is a graph and L is an on-labelling which indicates when vertices can be visited. An On-Hamiltonian walk is a sequence of all the vertices of G such that the position of each vertex modulo n is an integer of the label of that vertex. This paper will primarily investigate finding the shortest On-Hamiltonian walks in a blinking node system on complete graphs and complete bipartite graphs but also establishes the terminology and initial observations for working with blinking node systems on other graphs.

1 Introduction

Consider the situation where you wish to travel in a graph and visit every vertex; however, the vertices are labelled with discrete times for which they are able to be visited. Thus, a vertex cannot be visited at a time for which it is not labeled. This can be thought of as the vertices blinking "on" and "off" referring to when the vertex can be visited.

Problem (Traveling Baseball Fan)**.** *Imagine a baseball fan wishes to see a game in every baseball stadium. A stadium would be considered "on" if a game is played on that night in the stadium (the lights of that stadium are on). Natural questions arise about how one might travel to see a baseball game in every city. Other variations arise naturally where a city may be visited more than once.*

This is obviously not unique to our baseball analogy, as it seems natural to think of moving inside any network with restriction of when a node may be visited; for instance, virus scans on servers in use, a janitor cleaning rooms at a university while classes are in session, etc. So any scheduling problem that relies on an object travelling through a graph to "check" on every node, while being careful to visit nodes at allotted times can be viewed in this fashion. While many applications may also incorporate other such restrictions such as those discussed by travelling salesman [6], vehicle routing [2], and travelling purchaser problems [8], we wish to assume the baseball fan can travel from one stadium to the next before the next game takes place, without financial or time constraints.

Definition 1.1. *Let $B = (G, L)$ be a* ***blinking node system modulo*** *n, where G is a graph and $L : V(G) \to \mathcal{P}(\mathbb{Z}_n) \setminus \varnothing$ is a function known as an* ***on-labelling****. When the underlying graph is a complete graph on n vertices and the labellings are done modulo n, we denote the blinking node system as $BNS(n)$.*

The blinking node system shown in Figure 1 shows a walk $[a, b, g, i, d, c, h, j, e, a, f]$ highlighted in red which only "visits" a vertex when it is "on."

Observation 1. *If $n \neq |V(G)|$, it is usually possible to re-label the vertices modulo $|V(G)|$. For instance you can re-label vertex a in Figure 1 as $\{0, 2, 4, 6, 8\}$. However one needs to be careful doing so, as similarly relabelling vertex c to $\{0, 1, 3, 6, 7, 9\}$ may cause issues, because $10 \equiv 0 \pmod{10}$ but $10 \equiv 4 \pmod{6}$.*

While other variations may be interesting, the above observation leads us to focus our results on blinking node systems with modulo $|V(G)|$ in this paper.

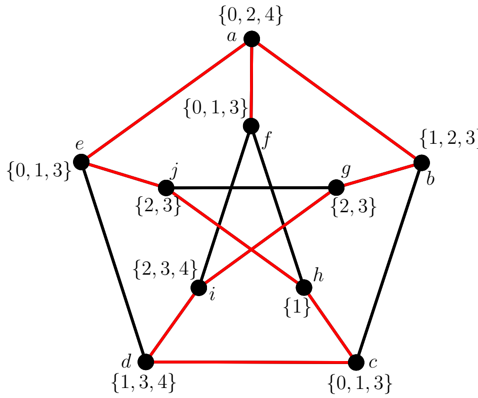


Figure 1: A Blinking Node System on the Petersen Graph modulo 5

Definition 1.2. *Let W be a walk of length $w-1$ represented by a sequence of w vertices in the graph G of a blinking node system. W is called an* ***on-walk*** *if for all $1 \leq i \leq w$ the vertex v_i, in the i^{th} position of W, $i-1 \pmod{n} \in L(v_i)$.*

More precisely we will be looking at on-walks that visit every vertex in G, which we will call **On-Hamiltonian walks** (not to be confused with Hamiltonian walks, which are closed). Furthermore an On-Hamiltonian walk of length $|V(G)|-1$ will be called an **On-Hamiltonian path**.

"Are there blinking node systems, where no On-Hamiltonian walk exists?" If the graph isn't connected, then obviously the BNS will not contain an On-Hamiltonian walk, but even blinking node systems on connected graphs can be made without On-Hamiltonian walks such as the one illustrated in Figure 2.

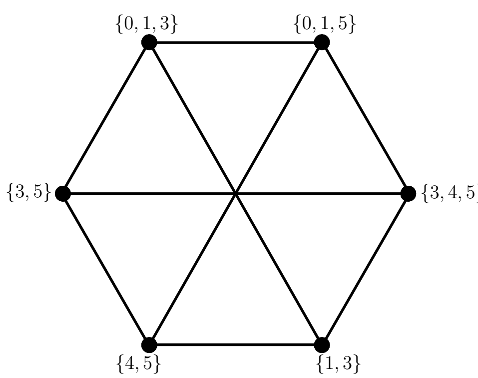


Figure 2: Graph with $2 \notin \bigcup_{v \in V(G)} L(v)$

Remark 1.1. *A necessary condition for the existence of an On-Hamiltonian walk in a blinking node system $B = (G, L)$ modulo n. For every $i \in \mathbb{Z}_n$ there exists some $v \in V(G)$ such that $i \in L(v)$. Otherwise stated as* $\bigcup_{v \in V(G)} L(v) = \mathbb{Z}_n$.

DOI: 10.20429/tag.2018.050102

One can find stronger necessary conditions for the existence of an On-Hamiltonian walk. For example, consider the set $S_i = \{v \mid i \in L(v)\}$; there must exist a vertex $w \in N(S_i)$ such that $i-1 \in L(w)$. If this does not hold, no on-walk will contain a neighbor of a vertex in S_i at time $i-1$, therefore the on-walk will not contain a vertex from S_i at time i, hence it is not an on-walk.

"How long can an On-Hamiltonian walk be?" Assuming there are no restrictions of having to visit a vertex when it is available before revisiting a vertex, these walks might be as long as one wishes. This is shown in the $BNS(5)$ of Figure 3 where the sequence $\{b, e, b, a, d, \ldots b, e, b, a, d, c\}$, such that the substring $\{b, e, b, a, d\}$ repeated as many times as desired, is an On-Hamiltonian walk.

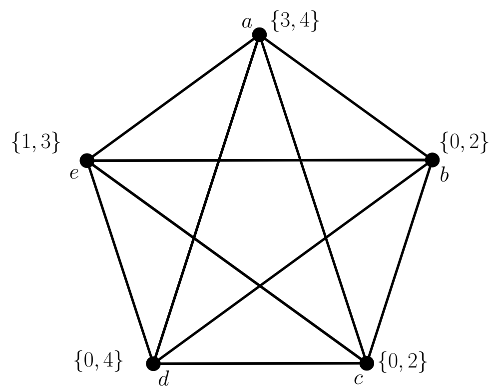


Figure 3: $BNS(5)$

The literature is rich on the many variations of the existence of Hamiltonian paths and cycles [4]. However, knowing if an undirected graph contains a Hamiltonian path is NP-complete [3], thus we will only explore blinking node systems for which the underlying graph is known to have a Hamiltonian path. We will start by considering $G = K_n$, for which any sequence of vertices is a walk, asking the general question: does a BNS(n) contain an on Hamiltonian walk?

Remark 1.1 states the necessity, but it is also sufficient that $\bigcup_{v \in V(G)} L(v) = \mathbb{Z}_n$ for an On-Hamiltonian walk to exist. One can achieve this by considering any arbitrary ordering of the vertices and greedily placing the vertices at the first unused time t, where $t \pmod{n} \in L(v)$ for every $v \in V(G)$. Now any unused time slots can be filled in by duplicating a vertex available at such a time; however, this may require visiting the same vertex for consecutive time slots. *We will discuss the idea of repeating a vertex later in Section 2 when searching for the shortest On-Hamiltonian walk for graphs that do not contain an On-Hamiltonian path.*

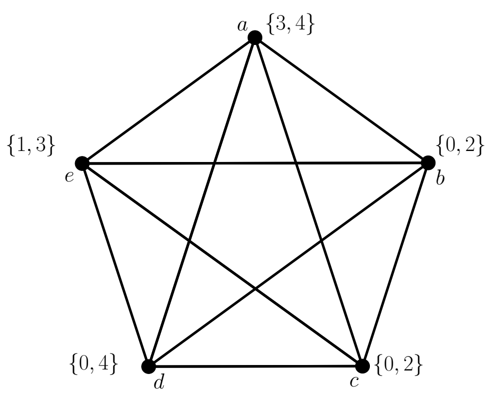
Observation 2. *If all but one vertex have the same labelling of size one, the On-Hamiltonian walk will be of length $2n-2$ which would be the longest such shortest On-Hamiltonian walk a BSN(n) could have.*

Since we have the necessary and sufficient condition for the existence of an On-Hamiltonian walk in BSN(n), it is natural to search for the shortest On-Hamiltonian walk in a BNS(n). We will begin by giving necessary and sufficient conditions for the existence of an On-Hamiltonian path in Theorem 2.2 and then generalize to find the shortest On-Hamiltonian walk in Theorem 2.3. Finally we extend these results for On-Hamiltonian walks in blinking node systems on complete bipartite graphs in Theorems 3.1 and 3.2 then applying all of these results to get Corrollary 3.3.

It should be noted that the conditions in Theorem 2.2 are similar to Hall's Theorem on a matching in a bipartite graph [5]. One can see this by constructing a bipartite graph $B(X,Y)$ with parts, $X = V(K_n)$ and $Y = \{x \mid 0 \leq x \leq n-1\}$ and edges, $E(B) = \{(x,y) \mid x \in X, y \in Y \text{ and } y \in L(x)\}$, represent the labelling of the blinking node system. Now applying Hall's Theorem to $B(X,Y)$ will determine if the blinking node system contains an On-Hamiltonian path. However this is highly dependant on the underlying graph of the blinking node system being complete, since $B(X,Y)$ neglects to model the edges of the graph in the blinking node system. Therefore we have provided a proof devoid of Hall's theorem in the spirit of generalizing this process to non-complete graphs. Furthermore since Hall's Theorem is typically proven by utilizing augmented paths of a matching via Berge's Theorem [1], one can see our proof is different since we begin with a bijection assigning all the vertices to a time slot and preform "switches" to find an On-Hamiltonian path.

2 Traveling in Complete Graphs

Let us begin with a small example on 5 vertices where each labelling is the same size. One could easily check all possible Hamiltonian paths in the graph in search of an On-Hamiltonian path, but for the purpose of our results, we illustrate an algorithmic approach of finding such an On-Hamiltonian path in Figure 4. The algorithm shown in Table 4b finds a vertex v, which is in a column numbered $x \notin L(v)$, then that vertex is "*switched*" with a vertex w, such that $x \in L(w)$. We formalize this "switching process" in Algorithm 2.1 and the proof of Theorem 2.2 shows that the algorithm will produce a sequence where the number of vertices in "incorrect" columns is 0 in a finite number of steps, hence producing on On-Hamiltonian path.



(a) BNS on K_5 with $|L(v)| = 2$

$step\backslash i$	0	1	2	3	4
1	a	b	c	d	e
2	c	b	a	d	e
3	c	a	b	d	e
4	c	e	b	d	a
5	c	e	b	a	d

(b) Algorithm 2.1 on Figure 4a

Figure 4: Finding an On-Hamiltonian path

DOI: 10.20429/tag.2018.050102

Algorithm 2.1.

0. Let $R : V(G) \to \mathbb{Z}_n$ be a bijection and $T = \emptyset$

1. While $v \in V(G)$ such that $R(v) \notin L(v)$

   (a) Find $w \in V(G)$, such that $(w, v) \notin T$ and $R(w) \in L(v)$

   (b) Define a permutation $\sigma = (R(w), R(v))$

   (c) Set $R = \sigma \circ R$ and $T = T \cup \{(w, v)\}$

   **Note that $(x, y) \circ R$ is a transposition of the vertices mapped to x and y respectively.*

2. Define the sequence H such that v is the $(R(v) + 1)^{st}$ term for all $v \in V(G)$.

3. Output H.

Since we are mostly concerned with vertices that are not mapped to integers in their labelling, let us define the following:

Definition 2.1. *Given a list $L(v)$ of the vertex v, the* ***complement*** $\overline{L(v)} = \mathbb{Z}_n \setminus L(v)$.

Definition 2.2. *Let* $\overline{R} = \left\{ v \in V(G) \mid R(v) \in \overline{L(v)} \right\}$

Theorem 2.2. *There exists an On-Hamiltonian path in a BNS(n) if and only if for every $S \in \mathcal{P}(\mathbb{Z}_n)$, $S \subseteq \overline{L(v)}$ for at most $n - |S|$ vertices of K_n.*

Proof. We will first show if the condition isn't met, no On-Hamiltonian path exists. Otherwise an On-Hamiltonian path is created by Algorithm 2.1.

For contradiction consider the negation, thus assuming there is a subset $S \in \mathcal{P}(\mathbb{Z}_n)$ such that $S \subseteq \overline{L(v)}$ where $n - |V| < |S|$. Therefore for all times i, where $i \in S$, an On-Hamiltonian path must use a vertex $w \notin V$; however, since $|S| > n - |V|$, there are not enough vertices to be mapped to the elements of S. Therefore by the pigeonhole principle, no such On-Hamiltonian path exists.

Now we assume that for every $S \in \mathcal{P}(\mathbb{Z}_n)$, $S \subseteq L(v)$ for at least $|S|$ vertices of K_n, and we will construct an On-Hamiltonian Path. This is equivalent to assuming if $Q \in \mathcal{P}(\mathbb{Z}_n)$ and $V = \{v \mid Q \subseteq \overline{L(v)}\}$, then $|Q| \leq n - |V|$. Algorithm 2.1 will suffice for constructing an On-Hamiltonian path.

When $|Q| = 1$ we have an equivalent statement to Remark 1.1, thus we see that whenever a vertex $v \in V(G)$, where $R(v) = i$ for $i \in \overline{L(v)}$, there exists a vertex w such that $i \in L(w)$. Furthermore, consider when $Q = \overline{L(v)}$, therefore there exists at least $n - |\overline{L(v)}|$ vertices which have a label containing an element in $L(v)$, but only $n - |\overline{L(v)}| - 1$ other elements $j \in \mathbb{Z}_n$ such that $j \in \overline{L(v)}$. Therefore there are enough vertices that can be transposed with v to guarantee that Algorithm 2.1 will create R such that $R(v) \in L(v)$. Since each transposition will map w to an element of $\mathbb{Z}_n$ within its labelling, no step of the algorithm will increase $|\overline{R}|$. However, once v is transposed such that $R(v) \in L(v)$ the algorithm will decrease $|\overline{R}|$.

Furthermore, keeping track of the transpositions with T provides that Step 1 of Algorithm 2.1 will terminate; therefore we have shown Algorithm 2.1 generates an On-Hamiltonian path in the BNS(n). □

"What is the shortest walk in a BNS where the conditions of Theorem 2.2 do not hold?" Let us say that S appears r times, where $r > n - |S|$. The proof of Theorem 2.2 gives us a natural lower bound of $n + r - |S|$ for the length of On-Hamiltonian walk. However, the labeling obviously provides more restrictions as shown in Figure 5, which illustrates a BNS(6) where the length of the shortest On-Hamiltonian walk is 8 ($[a, f, b, f, f, c, d, f, e]$) compared to stated lower bound of 7. Also, we can see that it may be necessary to visit a vertex in consecutive time slots, therefore we add loops at every vertex to illustrate the walk travelling along an edge when a vertex is adjacent to itself in the sequence of an On-Hamiltonian walk.

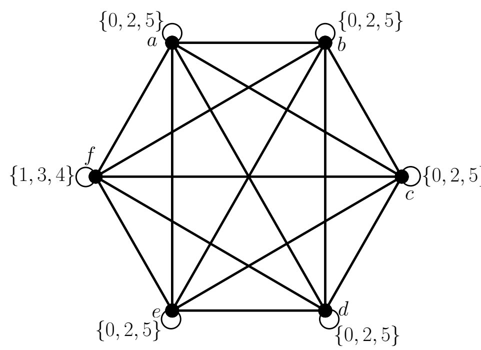


Figure 5: BNS(6) with loops

Let $S \subset \mathcal{P}(\mathbb{Z}_n)$ and $V_S = \{v \mid S \subseteq \overline{L(v)}\}$. If $|S| + |V_S| > n$, let $e_S = |S| + |V_S|$. Define $L_S = \bigcup_{v \in V_S} L(v)$ and $A_S = \{nx+i \mid i \in L_S \text{ and } x \in \mathbb{N}\}$. So for every $a \in A_S$, $a \pmod n \in L_S$, where $A_S = \{a_1, a_2, a_3 \dots\}$ such that $a_i < a_j$ whenever $i < j$. Finally, let $b_S = a_{e_S}$.

Theorem 2.3. *If Remark 1.1 is satisfied, then the length of the shortest On-Hamiltonian walk in the BNS(n) is* $l = \max\{b_S \mid S \in \mathcal{P}(\mathbb{Z}_n)\}$.

Since the only sets $S \in \mathcal{P}(\mathbb{Z}_n)$ that are requiring us to use more than n vertices are those where $|S| + |V_S| > n$ and whenever a superset of such a set also has this property, it will provide greater restrictions. We will define the set of deficiency time slots as

$$D = \{S \in \mathcal{P}(\mathbb{Z}_n) \mid n < |S| + |V_s| \text{ and } \forall X \text{ s.t. } S \subset X, |X| + |V_X| \leq n\}$$

Proof. We wish to build a one-to-one relation $R : V(G) \to \mathbb{N}_0$ similar to that described in Algorithm 2.1 for which each element of $V(G)$ is related to at least one element of $\mathbb{N}_0$ and the range is $\{x \mid 0 \leq x \leq t\}$. We can see that given an $S \in \mathcal{P}(\mathbb{Z}_n)$ the vertices in V_S need to be related to $|V_S|$ integers, for which b_S must be used. Consequently $t \geq l$ and thus any On Hamiltonian walk in the BNS will be at least this long.

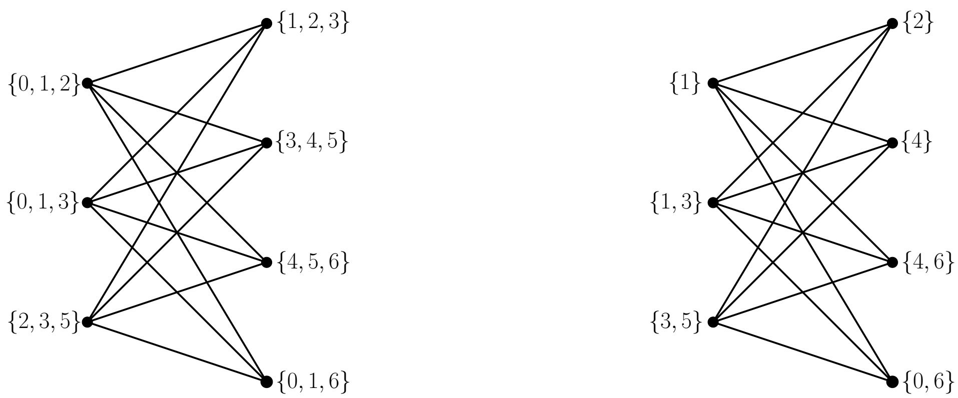
Thus it is sufficient to show that there exists a one-to-one relation R such that $t = l$, hence we will define $R : V(G) \to \mathbb{Z}_{l-1}$. Let $M = \{S \in D \mid b_S = l\}$ and $V_M = \bigcap_{S \in M} V_S$.

Define $R(v) = l - 1$ for some $v \in V_M$. The BNS$(n-1)$ on $G - v$ with the same on-labelling will satisfy $\max\{b_S \mid S \in \mathcal{P}(\mathbb{Z}_n)\} < l$. Thus recursively assign vertices of G to integers between n and $l-1$ until Theorem 2.2 is satisfied, at which point all remaining vertices can be assigned to positive integers less than n using Algorithm 2.1. Finally, for all $0 \leq i < l$ without an assigned vertex, we define $R(v) = i$ for some $v \in V(G)$ such that $i \in L(V)$, hence an On-Hamiltonian walk of length l exists where v is the $(R(v)+1)^{\text{st}}$ term for all $v \in V(G)$. □

3 Traveling in Complete Bipartite Graphs

We will denote a blinking node system on $K_{m,n}$ modulo $m+n$ as BNS(m,n). This section will examine similar arguments for complete bipartite graphs. Therefore assume the underlying graph of a BNS is $K_{m,n}$ with bi-partition X and Y.

The structure of bipartite graphs ensures that any walk will alternate between vertices in the set X and vertices in the set Y therefore the removal of any odd integer from $L(v)$ for v in the partition visited at even time intervals and any even integer from $L(w)$ for w in the other partition will not prohibit the existence of an On-Hamiltonian path. Figure 6 illustrates this removal process.



(a) Before Removal (b) After Removal

Figure 6: The Removal Process

Furthermore, we will view an on-walk as the alternation between two sequences; one sequence consisting only of vertices from X and the other consisting only of vertices from Y. Therefore it is natural to think of each partition as a complete graph needing an On-Hamiltonian path, but in such a way that only uses every other time interval, thus motivating the following definitions.

Definition 3.1. *Let $S \subseteq V(G)$ be a vertex partition of $K_{m,n}$. Define $B_S(even)$ to be a BSN$(|S|)$ on a complete graph on S with labelling M, where $i \in M(v)$ if and only if $2i \in L(v)$ for all $v \in S$.*

Definition 3.2. *Let $S \subseteq V(G)$ be a vertex partition of $K_{m,n}$. Define $B_S(odd)$ to be a BSN($|S|$) on a complete graph on S with labelling N, where $i \in N(v)$ if and only if $2i+1 \in L(v)$ for all $v \in S$.*

Consider a BSN(m, n) with bi-partition X, Y. Without loss of generality we can assume $|X| = m \geq n = |Y|$. By Ore's theorem [7] it is obvious that $|m - n| \leq 1$ is a necessary condition. Thus we need only consider two cases, either $m = n + 1$ or $m = n$.

Theorem 3.1. *A BSN$(n+1, n)$, B, with bi-partition X and Y such that $|X| = n+1$ and $|Y| = n$ has an On-Hamiltonian path if and only if $B_X(even)$ and $B_Y(odd)$ both contain On Hamiltonian paths*

Proof. Since any Hamiltonian path in $K_{n+1,n}$ must begin and end in X, we will consider B_X(even) and B_Y(odd). If B contains an On-Hamiltonian path H, then B_X(even) will contain an On-Hamiltonian path corresponding to the sequence of vertices visited by H at even indexed times and B_Y(odd) will contain an on Hamiltonian path corresponding to the sequence of vertices visited by H at odd indexed times.

If both B_X(even) and B_Y(odd) contain On-Hamiltonian paths, then the path constructed by alternating vertices from these two paths will be an On-Hamiltonian path in B. Therefore it is both necessary and sufficient that B_X(even) and B_Y(odd) satisfy the condition stated in Theorem 2.2 for B to contain an On-Hamiltonian path. □

Theorem 3.2. *A BSN(n, n), B, with bi-partition X and Y has an On-Hamiltonian path if and only if $B_X(even)$ and $B_Y(odd)$ both contain On Hamiltonian paths or $B_X(odd)$ and $B_Y(even)$ both contain On Hamiltonian paths.*

Proof. A Hamiltonian path in $K_{n,n}$ either begins in X and end in Y or begins in Y and ends in X, thus we recognize the possible need for both odd and even integers on the labelling of all vertices. However, without generality we can assume the On-Hamiltonian path starts in X and the same analysis of Theorem 3.1 applies, thus it is both necessary and sufficient that B_X(even) and B_Y(odd) both satisfy the condition stated in Theorem 2.2 for B to contain an On-Hamiltonian path starting on a vertex in X. □

Let B be a BSN(m, n) with bi-partitions X and Y. A similar removal can be used to find a shortest On-Hamiltonian walk, although care needs to be taken to consider the On-Hamiltonian walk starting in either partition, as in Theorem 3.2. Essentially we wish to apply Theorem 2.3 to both B_X(even) and B_Y(odd) or B_X(odd) and B_Y(even), depending on the sizes of X and Y.

To consider an On-Hamiltonian walk starting in X define x_1 to be the length of the shortest On-Hamiltonian walk in $B_X(even)$ and y_1 to be the length of the shortest On-Hamiltonian walk in $B_Y(odd)$. $l_1 = max\{2x_1, 2y_1 + 1\}$.Likewise, to consider an On-Hamiltonian walk starting in Y define x_2 to be the length of the shortest On-Hamiltonian walk in $B_X(odd)$ and y_2 to be the length of the shortest on Hamiltonian walk in $B_Y(even)$, so that $l_2 = max\{2x_2 + 1, 2y_2\}$.

Corollary 3.3. *The minimum On-Hamiltonian walk in $B = BSN(m, n)$ is $l = min\{l_1, l_2\}$.*

DOI: 10.20429/tag.2018.050102

4 Open Problems

As with any formalization of new language around a problem, there are many ways one can interpret the elements to ask new problems. In the case of on-walks in blinking node systems, there is the obvious questions about when do they exist or what conditions prohibit such a walk. Furthermore one could establish other criteria of the walks as we did with the On-Hamiltonian walk, such as an Eulerian version, but we will outline a few here that seem promising and are most related to what has been presented.

- **Finding On-Hamiltonian paths in known Hamiltonian graphs.** Keeping with the theme of the travelling baseball fan, it is natural to explore graphs which are known to be Hamiltonian. There are certainly characteristics, such as bridges, that reduce the search significantly, but we suggest looking at families of graphs which are highly connected such as complete multipartite graphs, polyhedral graphs, and grid graphs.

- **Finding On-Hamiltonian walks in Non-Hamiltonian graphs.** The graphs we considered were known to have a Hamiltonian path, but we then searched for the shortest On-Hamiltonian walk when Theorem 2.2 wasn't satisfied. Therefore it is natural to search for the shortest On-Hamiltonian walk in a BNS on a graph. For instance one can apply Corollary 3.3 to a Star graph, but what about arbitrary trees? What conditions are necessary for such an On-Hamiltonian walk to exist? Can we find the shortest On-Hamiltonian walk?

- **Finding shortest on-paths.** Consider the application of needing to travel from a start vertex to a target vertex in a BNS. Any on-walk starting and ending at the desired vertices would meet the application, but we would want to find the shortest such on-walk, which would naturally be called the shortest on-path. The first step would be to decide if such a path exists, then be able to find the shortest on-path. Again, we suspect the original inquiry should involve certain families of graphs or at least specific knowledge of the connectivity between the starting and target vertex.

References

[1] Berge, C.: Two theorems in graph theory. Proceedings of the National Academy of Sciences **43**(9), 842–844 (1957)

[2] Dantzig, G.B., Ramser, J.H.: The truck dispatching problem. Management science **6**(1), 80–91 (1959)

[3] Garey, M.R., Johnson, D.S., Stockmeyer, L.: Some simplified np-complete graph problems. Theoretical computer science **1**(3), 237–267 (1976)

[4] Gould, R.J.: Recent advances on the hamiltonian problem: Survey iii. Graphs and Combinatorics **30**(1), 1–46 (2014)

[5] Hall, P.: On representatives of subsets. Journal of the London Mathematical Society **1**(1), 26–30 (1935)

[6] Kruskal, J.B.: On the shortest spanning subtree of a graph and the traveling salesman problem. Proceedings of the American Mathematical society **7**(1), 48–50 (1956)

[7] Ore, O.: Note on hamilton circuits. The American Mathematical Monthly **67**(1), 55–55 (1960)

[8] Ramesh, T.: Traveling purchaser problem. Opsearch **18**(1-3), 78–91 (1981)

DOI: 10.20429/tag.2018.050102