On the Power of Waiting when Exploring Public Transportation Systems

David Ilcinkas, Ahmed Wade

On the Power of Waiting when Exploring Public Transportation Systems

David Ilcinkas[⋆] and Ahmed Mouhamadou Wade[⋆]

LaBRI, CNRS & Université de Bordeaux
{ilcinkas,wade}@labri.fr

Abstract. We study the problem of exploration by a mobile entity (agent) of a class of dynamic networks, namely the periodically-varying graphs (the PV-graphs, modeling public transportation systems, among others). These are defined by a set of carriers following infinitely their prescribed route along the stations of the network. Flocchini, Mans, and Santoro [FMS09] (ISAAC 2009) studied this problem in the case when the agent must always travel on the carriers and thus cannot wait on a station. They described the necessary and sufficient conditions for the problem to be solvable and proved that the optimal number of steps (and thus of moves) to explore a n-node PV-graph of k carriers and maximal period p is in $\Theta(k \cdot p^2)$ in the general case.
In this paper, we study the impact of the ability to wait at the stations. We exhibit the necessary and sufficient conditions for the problem to be solvable in this context, and we prove that waiting at the stations allows the agent to reduce the worst-case optimal number of moves by a multiplicative factor of at least $\Theta(p)$, while the time complexity is reduced to $\Theta(n \cdot p)$. (In any connected PV-graph, we have $n \leq k \cdot p$.) We also show some complementary optimal results in specific cases (same period for all carriers, highly connected PV-graphs). Finally this new ability allows the agent to completely map the PV-graph, in addition to just explore it.
Key words: Exploration, Dynamic graphs, Mobile agent, PV-graph

1 Introduction

1.1 The problem

The problem of graph exploration consists, for a mobile entity, in exploring all nodes (or edges) of an a priori unknown graph. This problem being one of the most classical in the mobile agent computing framework, it has received a lot of attention so far. Time complexity, space complexity or impact of a priori knowledge have extensively been studied in the last 40 years (see, e.g., [PP99,Rei05,DP04]). However, the large majority of these works concern static graphs. Considering nowadays networks, it is now common to deal with dynamic

[⋆] Partially supported by the ANR project ALADDIN, the INRIA project CEPAGE, and the European project EULER.

networks. In this paper, we study the graph exploration problem in one model of dynamic networks, namely the periodically-varying graph (PV-graph) model.

Roughly speaking, a PV-graph consists of a set of carriers, each following periodically its respective route among the sites of the system. This models in particular various types of public transportation systems like bus systems or subway systems for example. It also models low earth orbiting satellite systems, or security systems composed of security guards making tours in the place to be secured. Performing exploration in such systems may be useful for maintenance operations for example. Indeed, an agent can check that everything is in order during the exploration. This agent may be a piece of software, or a human being.

The exploration problem in the PV-graph model was already considered by Flocchini, Mans, and Santoro in [FMS09]. They considered that the agent cannot leave the carrier to stay on a site. Not being able to stay on a site is particularly legitimate in low earth orbiting satellite systems for example, where the sites do not correspond to any physical station. However, in most public transportation systems, it is possible for the agent (human or not) to stay on a site in order to wait for a (possibly different) carrier. In this paper, we consider the same problem but in the case when the agent can leave carriers to wait on a site. We study the impact of this new ability on the complexity (time and number of moves) of the PV-graph exploration problem.

1.2 Related work

Motivated by the automatic exploration of the Web, Cooper and Frieze [CF03] studied the question of the minimum cover time of a graph that evolves over time. They considered a particular model of so-called web graphs and show that if after every constant number of steps of the walk a new node appears and is connected to the graph, a random walk does not visit a constant fraction of nodes. Kuhn, Lynch and Oshman [KLO10] introduced a stability property (intervals of connectivity). They assume that for any T consecutive rounds, there is a stable and connected common subgraph. In 2008, Avin, Koucky and Lotker [AKL08] showed that a random walk may have an exponential cover time in some dynamic graphs. They also show that a variant, the lazy random walk, has however a polynomial cover time in any dynamic graph.

In 2009, Flocchini, Mans and Santoro [FMS09] introduced a new model of dynamic networks, the PV-graph model. They first show that if the nodes of the PV-graph are labeled, the knowledge of an upper bound on the longer period or the exact knowledge of the number n of nodes is necessary and sufficient for an agent to explore the PV-graph. If the nodes of the PV-graph are anonymous, then the knowledge of an upper bound on the longer period is necessary and sufficient. In both settings, the time and move complexity of the agent is proved to be in $\Theta(k \cdot p^2)$, where k is the number of carriers and p the maximum period of the carriers. In the particular case of homogeneous PV-graphs (PV-graphs for which all carriers have the same period), the time and move complexity drops to $\Theta(k \cdot p)$.

Flocchini, Kellett, Mason, and Santoro [FKMS10] studied the mapping of a PV-graph containing black holes (sites destroying agents). They considered that several agents are operating in the PV-graph, and that they can leave messages on the sites. The goal of the agents is to construct the map of the PV-graph without losing too many agents. Casteigts, Flocchini, Santoro and Quattrociocchi [CFQS10] integrated a large collection of concepts, formalisms and results in the literature about dynamic graphs in an unified space.

1.3 Our results

In this article, we extend the study of Flocchini, Mans and Santoro [FMS09] to the case when the agent can leave a carrier to stay at a site. This new ability allows the agent to explore PV-graphs that are less connected over time (formal definitions are given in Section 2). We prove that in the general case (so, even considering non highly-connected PV-graphs) the move complexity is reduced to $\Theta(\min\{k\cdot p, n\cdot p, n^2\})$, while the time complexity decreases to $\Theta(n\cdot p)$. (Note that in any connected PV-graph, we have $n \leq k\cdot p$.) If the PV-graphs are restricted to be both homogeneous and highly-connected, then Flocchini, Mans and Santoro proved that the time complexity is in $O(k \cdot p)$. In this paper, we prove that if the PV-graphs satisfy only one of these restrictions, then the time complexity remains in $\Theta(n \cdot p)$. Besides, it turns out that our algorithm not only performs exploration but also performs mapping, i.e., it can output an isomorphic copy of the PV-graph. Finally, note that our algorithm does not use possible identifiers of the nodes, while all our lower bounds still hold when the agent has access to unique node identifiers.

2 Model and definitions

We consider a system $S = \{s_1, \cdots, s_n\}$ of n *sites* among which k *carriers* are moving. Each carrier c has an identifier $\mathrm{Id}(c)$ and follows a finite sequence $R(c) = (s_{i_1}, \cdots, s_{i_{p(c)}})$ of sites, called its *route*, in a periodic manner. The positive integer $p(c)$ is called the *period* of the carrier c. More precisely, the carrier c starts at node s_{i_1} at time 0 and then proceeds along its route, moving to the next site at each time unit, in a cyclic manner (that is, when c is at node $s_{i_{p(c)}}$, it goes back to s_{i_1} and follows the route again and again).

A PV-graph (for periodically-varying graph) is a pair (S, C), where S is a set of sites, and C is a set of carriers operating among these sites. We will usually denote by n, k and p, respectively, the number of sites, the number of carriers and the maximum over the periods of the carriers. A PV-graph is said to be *homogeneous* if and only if all its carriers have the same period.

For any PV-graph G, we define two (classical) graphs $H_1(G)$ and $H_2(G)$ as follows. Both graphs have the set of carriers as the set of nodes. There is an edge in $H_1(G)$ between two carriers c and c' if and only if there exists a site appearing in the routes of c and c'. There is an edge in $H_2(G)$ between two carriers c and c' if and only if there exists a site s and a time $t \geq 0$ such that c and c' are both in s at time t. A PV-graph is said to be *connected* if and only if $H_1(G)$ is connected. A PV-graph is said to be *highly-connected* if and only if $H_2(G)$ is connected. In this paper, we will always consider PV-graphs that are at least connected. (Non-connected PV-graphs cannot be explored.) Furthermore note that, for any connected PV-graph, its parameters n (number of sites), k (number of carriers), and p (maximal period) satisfy the inequality $n - 1 \leq k(p - 1)$.

An entity, called *agent*, is operating on these PV-graphs. It can see the carriers and their identities. It can ride on a carrier to go from a site to another. Contrary to the model in [FMS09], the agent is allowed to leave a carrier, stay at the current site, and get back on a carrier (the same or another). We do not assume any restriction on the memory size of the agent or on its computational capabilities. We consider two models concerning the nodes' identities. In an *anonymous* PV-graph, the nodes do not have any identities, or the agent is not able to see them. In a *labeled* PV-graph, the nodes have distinct identities and the agent can see and memorize them.

We say that an agent *explores* a PV-graph if and only if, starting at time 0 on the starting site of the first carrier (this can be assumed without loss of generality), the agent eventually visits all sites of the PV-graph and switches afterwards to a terminal state. This terminal state expresses the fact that the agent knows that exploration has been completed.

3 Solvability

Similarly as in the case when the agent cannot wait, an agent without information on the PV-graphs it has to explore cannot explore all PV-graphs (even if restricted to the labeled homogeneous highly-connected ones).

Theorem 1. *There exists a family of labeled homogeneous highly-connected PV-graphs such that no agent can explore all the graphs of this family if it has no information on the PV-graphs it has to explore.*

Proof. Let $S = \{s_1, s_2, s_3\}$ be a set of three sites with distinct ids ($\mathrm{Id}(s_i) = i$). For $l > 0$, we define the PV-graph G_ℓ over the set S of sites composed of a single carrier. Its route is $(s_1, s_2, \cdots, s_1, s_2, s_1, s_3)$, where (s_1, s_2) is repeated exactly l times. Moreover, let G_0 be the PV-graph over the set of sites $\{s_1, s_2\}$ composed of a single carrier, whose route is (s_1, s_2). The family $\{G_0, G_1, \cdots\}$ is denoted $\mathcal{G}$.

Assume, for the purpose of contradiction, that there exists an algorithm solving the exploration problem in all the PV-graphs in $\mathcal{G}$, provided that the agent A running this algorithm does not receive any additional information. In particular, A explores G_0. Let m be the time at which A switches to the terminal state. Assume now that A is placed in G_m. For the first m time units, A cannot tell the difference between G_0 and G_m, because A has no information about the PV-graph it has to explore and in particular it does not know the number of sites or an upper bound on the system period. It will therefore act exactly the same in G_m than in G_0. In particular, it will switch to the terminal state at time m although the site s_3 has not yet been explored. This contradiction concludes the proof. □

4 General case

In this section, we make no assumption on the PV-graphs (except the connectedness assumption of course). We basically show that the ability to wait allows the agent to explore, and even map, all connected PV-graphs (not only the highly-connected ones), provided that the agent knows for each of them an upper bound on its maximal period. This can be done in only $\Theta(\min\{k \cdot p, n \cdot p, n^2\})$ moves, that is, at least p times less than when the agent cannot wait. Besides, the time complexity is reduced from $\Theta(k \cdot p^2)$ to $\Theta(n \cdot p)$.

4.1 Lower bound on the number of moves

Flocchini, Mans and Santoro [FMS09] proved a lower bound $\Omega(k \cdot p)$ on the number of moves to explore the PV-graphs with k carriers and maximum period p (even if restricted to the labeled homogeneous highly-connected ones). This lower bound does not apply directly in our setting because the agent, having the possibility to wait, could potentially be able to explore in significantly less moves. We will prove later that this is actually the case: the move complexity of our algorithm is bounded by $O(\min\{k \cdot p, n \cdot p, n^2\})$. We prove here that this complexity is optimal.

Lemma 1. *For any n, k, and p sufficiently large, $p \geq \lfloor \frac{n-1}{k} \rfloor + 1$ (necessary for connectedness), there exists a labeled homogeneous highly-connected PV-graph $G_{n,k,p}$ with n sites, k carriers and period p such that any algorithm needs at least* $\min\{k \cdot p - 1, \lfloor n/8 \rfloor \cdot p - 1, 7n/8 \cdot (\lfloor n/8 \rfloor - 1)\}$ *moves to explore it.*

Proof. Fix any integers $n \geq 8$, $k \geq 8$, and $p \geq 1$ such that $p \geq \lfloor \frac{n-1}{k} \rfloor + 1$. First assume that $k \leq n/8$.

- Subcase 1: $p \leq \frac{n^2}{4k} - k$.
  Let $q = \lfloor n/2k \rfloor$. Note that $q \geq 4$ and $p \geq q$. We denote by r the non-negative integer $\lceil p/q \rceil q - p$. Let $S = \{s_1, s_2, \ldots, s_n\}$ be a set of n sites. We partition S into the sets S_0 and $S_{i,j}$, with $1 \leq i \leq k$ and $1 \leq j \leq q$, such that:
  - $S_0 = \{s_1, s_2, \ldots, s_{\lceil p/q \rceil - 1}\}$ and $S_{1,1} = \{s_{\lceil p/q \rceil}\}$;
  - for all $1 \leq i \leq k$ and $1 \leq j \leq q$, we have $S_{i,j} \neq \emptyset$;
  - for all $2 \leq i \leq k$, we have $|S_{i,1}| \leq \lceil p/q \rceil - 1$;
  - for all $1 \leq i \leq k$ and $2 \leq j \leq q - r$, we have $|S_{i,j}| \leq \lceil p/q \rceil$;
  - for all $1 \leq i \leq k$ and $q - r < j \leq q$, we have $|S_{i,j}| \leq \lfloor p/q \rfloor$;

  Note that such a partition is always possible when p satisfies our assumption $\lfloor \frac{n-1}{k} \rfloor + 1 \leq p \leq \frac{n^2}{4k} - k$.
  The PV-graph $G_{n,k,p}$ is now defined as follows. Let S be its set of sites and $C = \{c_1, c_2, \ldots, c_k\}$ be the set of its carriers. For every $1 \leq i \leq k$, the route $R(c_i)$ is defined as follows. The route starts at s_1 at time 0 and then visits $s_2, s_3, \cdots, s_l$, with $l = \lceil p/q \rceil - |S_{i,1}|$, followed by each site of the set $S_{i,1}$. The route continues by visiting, for successive values of j from 2 to q, the sites $s_1, s_2, \cdots, s_l$, with $l = \lceil p/q \rceil - |S_{i,j}|$ (or $l = \lfloor p/q \rfloor - |S_{i,j}|$ if $j > q - r$),

followed by each site of the set $S_{i,j}$. Note that $G_{n,k,p}$ is both homogeneous (of period p) and highly-connected.

The PV-graph $G_{n,k,p}$ is constructed in such a way that the agent basically has to follow each carrier's route entirely to visit all sites. More precisely, to visit the sites of any set $S_{i,j}$, the agent has to pay $\lceil p/q \rceil$ moves ($\lfloor p/q \rfloor$ if $j > q - r$). Hence the minimum number of moves an exploring agent has to perform in $G_{n,k,p}$ is $k \cdot p - 1$.

– Subcase 2: $p > \frac{n^2}{4k} - k$.

Let us first assume that $k = \lfloor n/8 \rfloor$. The PV-graph $G_{n,k,p}$ is defined in this case as follows. Let $S = \{s_1, s_2, \ldots, s_n\}$ be the set of its sites and let $C = \{c_1, c_2, \ldots, c_k\}$ be the set of its carriers. For every $1 \leq i \leq \lfloor n/8 \rfloor$, the route $R(c_i)$ is any route of period p going through (and only through) sites $s_1, s_2, \ldots, s_{n-\lfloor n/8 \rfloor}$ and s_{n-i+1}, such that c_i is only once per period in s_{n-i+1}, just after being in $s_{n-\lfloor n/8 \rfloor}$, and just before being in s_1. Moreover, if c_i is in some site s_j, $2 \leq j \leq n - \lfloor n/8 \rfloor - 1$, at some time t, then at time $t+1$ the carrier c_i can only be at s_{j-1}, s_j, or s_{j+1}. We further assume that all carriers are in s_1 at time 0. If k is smaller than $\lfloor n/8 \rfloor$, then each carrier has to deal with several sites of the form s_{n-i+1}, with $1 \leq i \leq \lfloor n/8 \rfloor$. This is always possible thanks to our assumption on p. Note that $G_{n,k,p}$ is both homogeneous and highly-connected.

By construction, all sites s_{n-i+1}, with $1 \leq i \leq \lfloor n/8 \rfloor$, are only accessible through $s_{n-\lfloor n/8 \rfloor}$ and the agent can only leave them by going to s_1 with some carrier. Again by construction, any agent willing to go from s_1 to $s_{n-\lfloor n/8 \rfloor}$ has to go through all the sites $s_1, s_2, \ldots, s_{n-\lfloor n/8 \rfloor}$. Therefore, for any i, j such that $1 \leq i \neq j \leq \lfloor n/8 \rfloor$, going from s_{n-i+1} to s_{n-j+1} requires any agent to perform at least $n - \lfloor n/8 \rfloor + 1$ moves. Since any agent performing exploration of the PV-graph must visit all its sites, any agent requires at least $(n - \lfloor n/8 \rfloor + 1)(\lfloor n/8 \rfloor - 1)$ moves to explore $G_{n,k,p}$.

Now assume that $k > n/8$. In this case, we simply use the above constructions for $\lfloor n/8 \rfloor$ carriers. All carriers c_i, with $i > \lfloor n/8 \rfloor$ are given the same route as c_1 for example. This gives us immediately a lower bound $\lfloor n/8 \rfloor \cdot p - 1$ for $p \leq \frac{n^2}{4\lfloor n/8 \rfloor} - \lfloor n/8 \rfloor$ and still the lower bound $7n/8 \cdot (\lfloor n/8 \rfloor - 1)$ for $p > \frac{n^2}{4\lfloor n/8 \rfloor} - \lfloor n/8 \rfloor$. □

Summarizing the previous lemma by considering the asymptotic behavior, we directly obtain the following theorem.

Theorem 2. *The move complexity of the PV-graph exploration problem is in $\Omega(\min\{k \cdot p, n \cdot p, n^2\})$, where n, k, and p denote respectively the number of sites, the number of carriers, and the maximal period. This result holds even if the agent knows completely the PV-graph, has unlimited memory, and even in the labeled homogeneous highly-connected case.*

4.2 Lower bound on time

We can prove a larger lower bound for the time complexity than for the move complexity in the general case. More precisely, we have the following lemma.

Lemma 2. *Consider any n, k, and p, with $n \geq 6$, $2 \leq k \leq \frac{n-1}{3}$, and $p \geq \lfloor\frac{n-1}{k}\rfloor + 1$ (necessary for connectedness). There exists a family $\mathcal{G}_{n,p,k}$ of labeled homogeneous PV-graphs with n sites, k carriers and period p such that, for any algorithm, there exists a PV-graph in this family which cannot be explored by the algorithm using less than $(k-1)(p\lfloor\frac{n-1}{k}\rfloor - 1) + \lfloor\frac{n-1}{k}\rfloor - 1$ time steps.*

Proof. Fix any n, k, and p such that $n \geq 6$, $2 \leq k \leq \frac{n-1}{3}$, $p \geq \lfloor\frac{n-1}{k}\rfloor + 1$. Fix any $j_1, j_2, \ldots, j_{k-1}$ and $t_2, t_3, \ldots, t_k$ such that, for every $1 \leq i \leq k-1$, we have $(i-1)\lfloor\frac{n-1}{k}\rfloor + 2 \leq j_i \leq i\lfloor\frac{n-1}{k}\rfloor + 1$ and $1 \leq t_{i+1} \leq p$. The PV-graph $G((j_1, t_2), (j_2, t_3), \ldots, (j_{k-1}, t_k))$ is defined as follows.

Let $S = \{s_1, s_2, \ldots, s_n\}$ be the set of its sites and let $C = \{c_1, c_2, \ldots, c_k\}$ be the set of its carriers. Let us partition S into $k+1$ subsets $S_0, S_1, \ldots, S_{k-1}, S_k$ such that $S_0 = \{s_1\}$, $S_i = \{s_{(i-1)\lfloor\frac{n-1}{k}\rfloor+2}, \ldots, s_{i\lfloor\frac{n-1}{k}\rfloor+1}\}$, for $1 \leq i \leq k-1$, and S_k contains all the remaining sites.

Let $j_0 = 1$ and $t_1 = 0$. Consider any i such that $1 \leq i \leq k$. The route $R(c_i)$ is any route of period p going through (and only through) all the sites in $S_i \cup \{s_{j_{i-1}}\}$ satisfying the following two conditions. First, c_i visits $s_{j_{i-1}}$ only once per period, at all times equal to t_i modulo p. Second, the route $R(c_i)$ does not depend on the values j_l and t_{l+1}, for $l \geq i$.

The family $\mathcal{G}_{n,p,k}$ is defined as the set of all PV-graphs $G((j_1, t_2), (j_2, t_3), \ldots, (j_{k-1}, t_k))$ with, for every $1 \leq i \leq k-1$, $(i-1)\lfloor\frac{n-1}{k}\rfloor + 2 \leq j_i \leq i\lfloor\frac{n-1}{k}\rfloor + 1$ and $1 \leq t_{i+1} \leq p$. All these PV-graphs are labeled homogeneous PV-graphs with n sites, k carriers and period p.

Let A be any exploring agent (i.e. executing any exploration algorithm). Given $1 \leq i \leq k$ and G a PV-graph of $\mathcal{G}_{n,p,k}$, let $\mathcal{T}_i(G)$ be the first time at which the agent A, starting at s_1 at time 0 in G, sees the carrier c_i. Given q, $1 \leq q \leq k$, and $j_1, j_2, \ldots, j_{q-1}$ and $t_2, t_3, \ldots, t_q$ in the usual ranges, we define $\mathcal{G}_{n,p,k}((j_1, t_2), (j_2, t_3), \ldots, (j_{q-1}, t_q))$ as the set of all the PV-graphs $G((j_1, t_2), (j_2, t_3), \ldots, (j_{k-1}, t_k))$ with, for every $q \leq i \leq k-1$, $(i-1)\lfloor\frac{n-1}{k}\rfloor + 2 \leq j_i \leq i\lfloor\frac{n-1}{k}\rfloor + 1$ and $1 \leq t_{i+1} \leq p$.

Claim. For every q, $1 \leq q \leq k$, and every i, $1 \leq i \leq q-1$, there exist j_i and t_{i+1} satisfying $(i-1)\lfloor\frac{n-1}{k}\rfloor + 2 \leq j_i \leq i\lfloor\frac{n-1}{k}\rfloor + 1$ and $1 \leq t_{i+1} \leq p$ such that for every graph $G \in \mathcal{G}_{n,p,k}((j_1, t_2), (j_2, t_3), \ldots, (j_{q-1}, t_q))$ we have $\mathcal{T}_q(G) \geq (q-1)(p\lfloor\frac{n-1}{k}\rfloor - 1)$.

Proof of the Claim: We prove the claim by induction on q. The base case $q = 1$ is trivially true. Fix any q such that $1 \leq q \leq k-1$, and assume, by induction hypothesis, that the claim holds for the value q.

Let $\mathcal{G}_q$ be the family $\mathcal{G}_{n,p,k}((j_1, t_2), (j_2, t_3), \ldots, (j_{q-1}, t_q))$ whose existence is guaranteed by the induction hypothesis. Note that all PV-graphs in $\mathcal{G}_q$ have exactly the same routes $R(c_i)$, for $1 \leq i \leq q$. We can thus define H_q to be the PV-graph consisting only of the carriers c_1 to c_q of any PV-graph in $\mathcal{G}_q$. Let us consider now the agent A starting at s_1 at time 0 in H_q. By induction hypothesis and by construction of H_q, the agent A sees c_q for the first time at time t with $t \geq (q-1)(p\lfloor\frac{n-1}{k}\rfloor - 1)$ time steps. Thus there exists j_q and t_{q+1} satisfying $(q-1)\lfloor\frac{n-1}{k}\rfloor + 2 \leq j_q \leq q\lfloor\frac{n-1}{k}\rfloor + 1$ and $1 \leq t_{q+1} \leq p$ such that A is never at

s_{j_q} at a time equal to t_{q+1} modulo p before time $t + p\lfloor\frac{n-1}{k}\rfloor - 1$, and thus before time $q(p\lfloor\frac{n-1}{k}\rfloor - 1)$.

Consider now the agent A starting at s_1 at time 0 in any PV-graph G in $\mathcal{G}_{n,p,k}((j_1, t_2), (j_2, t_3), \dots, (j_{q-1}, t_q), (j_q, t_{q+1}))$. Before time $q(p\lfloor\frac{n-1}{k}\rfloor - 1)$, the agent will behave exactly the same as in H_q and will not see the carrier c_{q+1}. This concludes the proof of the claim. $\diamondsuit$

The theorem now follows by considering the claim for the last value $q = k-1$, and by noting that the agent still has to visit all sites of S_k after reaching c_k, which requires additional $\lfloor\frac{n-1}{k}\rfloor - 1$ time steps. $\square$

Again, summarizing the previous lemma by considering the asymptotic behavior, we directly obtain the following theorem.

Theorem 3. *The time complexity of the PV-graph exploration problem is in $\Omega(n \cdot p)$ in the general case. This result holds even if the agent knows n, k, and p, has unlimited memory, and even in the labeled case.*

4.3 Our algorithm

In the above part of the paper, we exhibited some necessary conditions on the existence of a solution. We then provided lower bounds on the move and time complexities. We now essentially prove that all these results are optimal by describing and proving a PV-graph exploration algorithm with matching upper bounds on the move and time complexities, provided that the agent knows a linear upper bound B on the maximum period p. As a consequence, we show that the ability to wait allows to decrease both the move and time complexities, the former by a multiplicative factor at least $\Theta(p)$.

Algorithm EXPLORE-WITH-WAIT

Our algorithm stores a matrix Mat where lines correspond to (known so far) carriers. The algorithm progressively fills in each line with the sequence of sites visited by the corresponding carrier. In order to do that, the agent stays $2B$ steps at each site, looking at each visit of the carriers at this site. From each partially filled in line, the algorithm computes a divisor of the period of the corresponding carrier, allowing the agent to predict the exact schedule of the carriers at the sites already known by the agent. The algorithm also maintains a tree of carriers, where a carrier c is a child of a carrier c' if c was discovered for the first time while visiting c'. The algorithm visits successively new sites until the whole matrix is filled in. Note that the completed matrix contains the complete schedule of all carriers. Hence one can easily extract a map of the PV-graph from the matrix.

Let a start with carrier c_1. Initially: $Home = c_1$; $parent(Home) := \emptyset$; $Visited := \emptyset$; $ToExplore := \{c_1\}$; $p := 1$.

Algorithm 1 EXPLORE-WITH-WAIT (c)

```
if $c = Home$ and $ToExplore = \emptyset$ then
    Terminate
else
    if $c \notin Visited$ then
        VISIT(c)
    end if
    $c' \leftarrow$ NEXT(c)
    EXPLORE-WITH-WAIT (c’)
end if
```

Algorithm 2 NEXT-EMPTY-CELL (Mat, j, c, v)

```
$p =$ period($Mat[idC]$)
while $Mat[Id(c), j] \neq v$ and $j < B$ do
    $u \leftarrow Mat[Id(c), j]$
    $i \leftarrow j + 1$
    while $Mat[Id(c), i] \neq u$ and $Mat[Id(c), i] \neq v$ and $i < B$ do
        $i \leftarrow i + 1$
    end while
    if Mat[Id(c), i]=u then
        Get on $c$ at its next visit at the current site $u$
        $j \leftarrow i$
    else
        if $Mat[Id(c), i] = v$ then
            Get on $c$ at the first time $k$ such that $k$ mod $p = j$
            Do one move with $c$
            $j \leftarrow j + 1$
            if $Mat[Id(c), j] \neq v$ then
                Get off on the current site
            end if
        else
            $j \leftarrow i$
        end if
    end if
end while
Return $j$
```

Algorithm 3 NEXT(c)

```
if $(N(c)) = \emptyset$ and $(c = c_0)$ then
    Return $c_0$
else if $c \neq c_0$ then
    $c' \leftarrow NEXT(parent(c))$
else
    $c' \leftarrow$ an element of $N(c) \cap ToExplore$
end if
$p =$ period($Mat[Id(c')]$)
$i \leftarrow 0$
while $Mat[Id(c'), i] == \emptyset$ do
    $i \leftarrow i + 1$
end while
v=Mat[Id(c'), i]
$j \leftarrow$ NEXT-EMPTY-CELL $(Mat, j, c, v)$
Stay on site $v$
Get on $c'$ at the first time $k$ such that $k \bmod p = i$
$Parent$(c') $:= c$
Return $c'$
```

Algorithm 4 VISIT (c, j)

```
$MyParent \leftarrow$ parent $(c)$; $N(c) := \{MyParent\}$
$i \leftarrow j$
while $j < B + i$ do
    $u \leftarrow$ current site
    Get off on site $u$
    if $Mat[Id(c), j] == \emptyset$ then
        $Mat[Id(c), j] \leftarrow u$
        while $i < (2B + j)$ do
            Stay on $u$ and at each step DO
            $i \leftarrow i + 1$
            if $c$ visits $u$ at this step then
                $Mat[Id(c), i \bmod 2B] \leftarrow u$
            else
                if the agent sees $c' \notin (ToExplore \cap Visit)$ then
                    $Mat[Id(c'), i \bmod 2B] \leftarrow u$
                    $ToExplore := ToExplore \cup$ {c'}
                    $N(c) := N(c) \cup \{c'\}$
                end if
            end if
            $p =$ period($Mat[Id(c)]$)
        end while
    else
        $j \leftarrow \mathtt{NEXT-EMPTY-CELL(Mat, j, c, \emptyset)}$
    end if
end while
$Visit \leftarrow Visit \cup \{c\}$
$ToExplore \leftarrow ToExplore - \{c\}$
```

Correctness

Theorem 4. *Algorithm EXPLORE-WITH-WAIT correctly explores and maps in finite time any PV-graph, even anonymous, but provided that an upper bound on the maximum period is known.*

Proof. First observe that when an agent stays at a site for $2B$ steps, where B is the known upper bound on the maximum period, it sees all the carriers visiting that site. Moreover, after filling in the matrix with that information, it is able to predict at any point in the future which carrier will be at that site. Since the PV-graph is connected, the agent will miss no carriers and thus no sites either. At the end of the algorithm, the matrix will be completely filled in and it will be equivalent to a map of the PV-graph. □

Move and time complexities

Theorem 5. *With the algorithm EXPLORE-WITH-WAIT, the agent makes at most $O(\min\{k \cdot p, n \cdot p, n^2\})$ moves to explore any n-site k-carrier PV-graph of maximum period p.*

Proof. Let us first prove that the move complexity is in $O(n^2)$. Obviously, the agent only moves when looking for the next empty cell. Since an empty cell always corresponds to a new unvisited site, looking for the next empty cell is done at most n times. The algorithm is done in such a way that, during the travel from the last visited site u to the following new site v, each site w is visited at most once. Indeed, it is always possible for the agent to wait on w for the appropriate carrier to come at w. Hence the number of moves is bounded by n^2.

We now prove that the move complexity is in $O(k \cdot p)$. During a single travel to go to the next empty cell, the agent may have to use several carriers. However, we visit the carriers following a DFS traversal of the tree of carriers. Hence in total the agent uses at most $2k$ carriers. When using a carrier, the agent does at most p moves. Hence the number of moves is bounded by $2k \cdot p$.

We finally prove that the move complexity is in $O(n \cdot p)$. This is done by refining the previous argument. A carrier is always added as a leaf to the tree of carriers. Moreover, a carrier is used only if the agent goes to an empty cell of this carrier. Since the agent goes to at most n empty cells, it means that at most n carriers of the tree are used. Hence the number of moves is bounded by $2n \cdot p$. □

Theorem 6. *The algorithm EXPLORE-WITH-WAIT allows to explore any n-node PV-graph in $O(nB)$ time steps, where B is a known upper bound on p.*

Proof. A lot of time is spent by the agent by staying $O(B)$ steps on a site to note all passing carriers. Since there are n sites to visit, the agent spends at most $O(nB)$ time steps doing this. It turns out that this is the main cost of the algorithm in terms of time complexity. Indeed, as noticed in the previous proof, the agent uses at most $2\min\{k,n\}$ carriers when traveling. On each carrier, the agent uses not only at most p moves but also at most p time steps. Hence the completion time of the algorithm is at most $O(nB)+2\min\{k,n\}\cdot p$. This proves the theorem. □

As noticed before, we have the following corollary.

Corollary 1. *Given the a priori knowledge of an upper bound $B = O(p)$ on the maximum period p, Algorithm $EXPLORE-WITH-WAIT$ is asymptotically optimal in the general case with respect to both the move and the time complexities. The optimal move complexity is in $\Theta(\min\{k\cdot p, n\cdot p, n^2\})$ while the optimal time complexity is in $\Theta(n\cdot p)$.*

5 Specific cases

We showed in the previous section the optimal move and time complexities for the PV-graph exploration problem in the general case. This section is devoted to the specific cases of homogeneous or highly-connected PV-graphs. In both cases, we prove that the move and time complexities remain the same as in the general case. Note, however, that when considering PV-graphs being both homogeneous and highly-connected, we know from [FMS09] that the optimal time complexity is at most $O(k\cdot p)$, even when n is large.

5.1 The homogeneous case

If we consider the homogeneous PV-graphs (but possibly not highly-connected), the time and move complexities remain the same as in the general case.

Theorem 7. *Given the a priori knowledge of an upper bound $B = O(p)$ on the maximum period p, Algorithm $EXPLORE-WITH-WAIT$ is asymptotically optimal in the homogeneous case with respect to both the move and the time complexities. The optimal move complexity is in $\Theta(\min\{k\cdot p, n\cdot p, n^2\})$ while the optimal time complexity is in $\Theta(n\cdot p)$.*

Proof. The result directly follows from Lemma 2 and Corollary 1. □

5.2 The highly-connected case

If we consider the highly-connected PV-graphs (but possibly not homogeneous), the time and move complexities remain the same as in the general case.

Lemma 3. *Consider any n, k, and p, with $n \geq 6$, $2 \leq k \leq \frac{n-1}{3}$, $p \geq \lfloor\frac{n-1}{k}\rfloor + 2$. There exists a family $\mathcal{G}'_{n,p,k}$ of labeled highly-connected PV-graphs with n sites, k carriers and maximum period p such that, for any algorithm, there exists a PV-graph in this family which cannot be explored by the algorithm using less than $\lceil\frac{k-1}{2}\rceil(p\lfloor\frac{n-1}{k}\rfloor - 1) + \lfloor\frac{k-1}{2}\rfloor((p-1)\lfloor\frac{n-1}{k}\rfloor - 1) + \lfloor\frac{n-1}{k}\rfloor - 1$ time steps.*

Proof. Fix any n, k, and p such that $n \geq 6$, $2 \leq k \leq \frac{n-1}{3}$, $p \geq \lfloor \frac{n-1}{k} \rfloor + 2$. Fix any $j_1, j_2, \ldots, j_{k-1}$ and $t_2, t_3, \ldots, t_k$ such that, for every $1 \leq i \leq k-1$, we have $(i-1)\lfloor \frac{n-1}{k} \rfloor + 2 \leq j_i \leq i\lfloor \frac{n-1}{k} \rfloor + 1$ and $1 \leq t_{i+1} \leq p$, if i is odd, $1 \leq t_{i+1} \leq p-1$, if i is even. The PV-graph $G((j_1, t_2), (j_2, t_3), \ldots, (j_{k-1}, t_k))$ is defined as follows.

Let $S = \{s_1, s_2, \ldots, s_n\}$ be the set of its sites and let $C = \{c_1, c_2, \ldots, c_k\}$ be the set of its carriers. Let us partition S into $k+1$ subsets $S_0, S_1, \ldots, S_{k-1}, S_k$ such that $S_0 = \{s_1\}$, $S_i = \{s_{(i-1)\lfloor \frac{n-1}{k} \rfloor + 2}, \ldots, s_{i\lfloor \frac{n-1}{k} \rfloor + 1}\}$, for $1 \leq i \leq k-1$, and S_k contains all the remaining sites.

Let $j_0 = 1$ and $t_1 = 0$. Consider any i such that $1 \leq i \leq k$. The route $R(c_i)$ is any route going through (and only through) all the sites in $S_i \cup \{s_{j_{i-1}}\}$ satisfying the following three conditions. First, c_i is of period $p-1$ if i is odd, and of period p if i is even. Second, c_i visits $s_{j_{i-1}}$ only once per period, at all times equal to t_i modulo its period. Third, the route $R(c_i)$ does not depend on the values j_l and t_{l+1}, for $l \geq i$.

The family $\mathcal{G}'_{n,p,k}$ is defined as the set of all PV-graphs $G((j_1, t_2), (j_2, t_3), \ldots, (j_{k-1}, t_k))$ with, for every $1 \leq i \leq k-1$, $(i-1)\lfloor \frac{n-1}{k} \rfloor + 2 \leq j_i \leq i\lfloor \frac{n-1}{k} \rfloor + 1$ and $1 \leq t_{i+1} \leq p$, if i is odd, $1 \leq t_{i+1} \leq p-1$, if i is even. All these PV-graphs are labeled highly-connected PV-graphs with n sites, k carriers and maximum period p. (Indeed, note that, for every $1 \leq i \leq k-1$, c_i and c_{i+1} meets at s_{j_i} at most every $p(p-1)$ steps.)

Let A be any exploring agent (i.e. executing any exploration algorithm). Given $1 \leq i \leq k$ and G a PV-graph of $\mathcal{G}'_{n,p,k}$, let $\mathcal{T}_i(G)$ be the first time at which the agent A, starting at s_1 at time 0 in G, sees the carrier c_i. Given q, $1 \leq q \leq k$, and $j_1, j_2, \ldots, j_{q-1}$ and $t_2, t_3, \ldots, t_q$ in the usual ranges, we define $\mathcal{G}'_{n,p,k}((j_1, t_2), (j_2, t_3), \ldots, (j_{q-1}, t_q))$ as the set of all the PV-graphs $G((j_1, t_2), (j_2, t_3), \ldots, (j_{k-1}, t_k))$ with, for every $q \leq i \leq k-1$, $(i-1)\lfloor \frac{n-1}{k} \rfloor + 2 \leq j_i \leq i\lfloor \frac{n-1}{k} \rfloor + 1$ and $1 \leq t_{i+1} \leq p$, if i is odd, $1 \leq t_{i+1} \leq p-1$, if i is even.

Claim. For every q, $1 \leq q \leq k$, and every i, $1 \leq i \leq q-1$, there exist j_i and t_{i+1} satisfying $(i-1)\lfloor \frac{n-1}{k} \rfloor + 2 \leq j_i \leq i\lfloor \frac{n-1}{k} \rfloor + 1$ and $1 \leq t_{i+1} \leq p$ ($t_{i+1} \leq p-1$ when i is even) such that for every graph $G \in \mathcal{G}'_{n,p,k}((j_1, t_2), (j_2, t_3), \ldots, (j_{q-1}, t_q))$ we have $\mathcal{T}_q(G) \geq \lceil \frac{q-1}{2} \rceil (p\lfloor \frac{n-1}{k} \rfloor - 1) + \lfloor \frac{q-1}{2} \rfloor ((p-1)\lfloor \frac{n-1}{k} \rfloor - 1)$.

Proof of the Claim: We prove the claim by induction on q. The base case $q = 1$ is trivially true. Fix any q such that $1 \leq q \leq k-1$, and assume, by induction hypothesis, that the claim holds for the value q.

Let $\mathcal{G}'_q$ be the family $\mathcal{G}'_{n,p,k}((j_1, t_2), (j_2, t_3), \ldots, (j_{q-1}, t_q))$ whose existence is guaranteed by the induction hypothesis. Note that all PV-graphs in $\mathcal{G}'_q$ have exactly the same routes $R(c_i)$, for $1 \leq i \leq q$. We can thus define H'_q to be the PV-graph consisting only of the carriers c_1 to c_q of any PV-graph in $\mathcal{G}'_q$. Let us consider now the agent A starting at s_1 at time 0 in H'_q. By induction hypothesis and by construction of H'_q, the agent A sees c_q for the first time at time t with $t \geq \lceil \frac{q-1}{2} \rceil (p\lfloor \frac{n-1}{k} \rfloor - 1) + \lfloor \frac{q-1}{2} \rfloor ((p-1)\lfloor \frac{n-1}{k} \rfloor - 1)$ time steps. Thus there exists j_q and t_{q+1} satisfying $(q-1)\lfloor \frac{n-1}{k} \rfloor + 2 \leq j_q \leq q\lfloor \frac{n-1}{k} \rfloor + 1$ and $1 \leq t_{q+1} \leq p$, if q is odd, $1 \leq t_{q+1} \leq p-1$, if q is even, such that A is never at s_{j_q} at a time equal to t_{q+1} modulo the period p' of c_{q+1} before time $t + p'\lfloor\frac{n-1}{k}\rfloor - 1$, and thus before time $\lceil\frac{q}{2}\rceil(p\lfloor\frac{n-1}{k}\rfloor - 1) + \lfloor\frac{q}{2}\rfloor((p-1)\lfloor\frac{n-1}{k}\rfloor - 1)$.

Consider now the agent A starting at s_1 at time 0 in any PV-graph G in $\mathcal{G}'_{n,p,k}((j_1, t_2), (j_2, t_3), \ldots, (j_{q-1}, t_q), (j_q, t_{q+1}))$. Before time $\lceil\frac{q}{2}\rceil(p\lfloor\frac{n-1}{k}\rfloor - 1) + \lfloor\frac{q}{2}\rfloor((p-1)\lfloor\frac{n-1}{k}\rfloor - 1)$, the agent will behave exactly the same as in H'_q and will not see the carrier c_{q+1}. This concludes the proof of the claim. $\lozenge$

The theorem now follows by considering the claim for the last value $q = k-1$, and by noting that the agent still has to visit all sites of S_k after reaching c_k, which requires additional $\lfloor\frac{n-1}{k}\rfloor - 1$ time steps. $\square$

Again, summarizing the previous lemma, using Corollary 1, and considering the asymptotic behavior, we obtain the following theorem.

Theorem 8. *Given the a priori knowledge of an upper bound $B = O(p)$ on the maximum period p, Algorithm $EXPLORE-WITH-WAIT$ is asymptotically optimal in the highly-connected case with respect to both the move and the time complexities. The optimal move complexity is in $\Theta(\min\{k \cdot p, n \cdot p, n^2\})$ while the optimal time complexity is in $\Theta(n \cdot p)$.*

References

[AKL08] C. Avin, M. Koucky, and Z. Lotker. How to explore a fast-changing world (cover time of a simple random walk on evolving graphs). In *35th International Colloquium on Automata, Languages and Programming (ICALP)*, LNCS 5125, pages 121–132, 2008.

[CFQS10] A. Casteigts, P. Flocchini, W. Quattrociocchi, and N. Santoro. Time-varying graphs and dynamic networks. *CoRR*, abs/1012.0009, 2010.

[CF03] C. Cooper and A. M. Frieze. Crawling on simple models of web graphs. *Internet Mathematics*, 1(1), 2003.

[DP04] A. Dessmark and A. Pelc. Optimal graph exploration without good maps. *Theor. Comput. Sci.*, 326(1-3), pages 343–362, 2004.

[FKMS10] P. Flocchini, M. Kellett, P. C. Mason, and N. Santoro. Mapping an unfriendly subway system. In *5th Intl Conference on Fun with Algorithms (FUN)*, LNCS 6099, pages 190–201, 2010.

[FMS09] P. Flocchini, B. Mans, and N. Santoro. Exploration of periodically varying graphs. In *20th Intl Symposium on Algorithms and Computation (ISAAC)*, LNCS 5878, pages 534–543, 2009.

[KLO10] F. Kuhn, N. A. Lynch, and R. Oshman. Distributed computation in dynamic networks. In *42nd ACM Symposium on Theory of Computing (STOC)*, pages 513–522, 2010.

[PP99] P. Panaite and A. Pelc. Exploring Unknown Undirected Graphs. *J. Algorithms*, 33(2), pages 281–295, 1999.

[Rei05] O. Reingold. Undirected st-connectivity in log-space. In *37th ACM Symposium on Theory of Computing (STOC)*, pages 376–385, 2005.