



Finding Colorful Paths in Temporal Graphs

Riccardo Dondi^(✉) and Mohammad Mehdi Hosseinzadeh

Università degli Studi di Bergamo, Bergamo, Italy
{riccardo.dondi,m.hosseinzadeh}@unibg.it

Abstract. The problem of finding paths in temporal graphs has been recently considered due to its many applications. In this paper we consider a variant of the problem that, given a vertex-colored temporal graph, asks for a path whose vertices have distinct colors and include the maximum number of colors. We study the approximation complexity of the problem and we provide an inapproximability lower bound. Then we present a heuristic for the problem and an experimental evaluation of our heuristic, both on synthetic and real-world graphs.

Keywords: Temporal graphs · Algorithms on graphs · Algorithms for network analysis · Heuristics · Approximation complexity

1 Introduction

Finding paths is a basic problem in graph theory [9] and several variants have been studied, including finding a shortest path between two vertices and finding a longest path in a graph. Recently, these problems have been considered for real-world data that need a description of the vertex properties and dynamics of the relations [19]. For these data, a richer representation with respect to the classical graph model has to be introduced, for example by associating labels or colors with vertices and by representing the evolution of relations with a temporal graph. In this latter model, edges are associated with timestamps to represent when an interaction occurred [6, 10, 15].

In this paper we consider a problem that looks for a path in a temporal graph that has vertices associated with colors. Given a set of colors, the problem asks for a temporal path having vertices with distinct colors and including the maximum number of colors. A temporal path in a temporal graph is a path in which the timestamps of consecutive edges are strictly increasing, thus representing a path that does not violate the time constraint specified by the timestamps of the edges. The problem we consider is a variant of the one considered in [19], that asks for a temporal path that exactly matches a multiset of colors (called motif in [19]). As outlined in [19], this problem has several applications, for example in tour recommendations [7, 13], where vertices correspond to interesting locations, colors represent activities available in locations, edges correspond to

transportation links between different locations (a timestamp is associated for example to departure time). A set (or a multiset) of colors represents activities a tourist may be interested into, and a path associated with different colors is then a suggestion of the activities that can be carried out respecting the time constraints. A temporal graph, due to its structure, may not contain a temporal path that includes all the colors. Thus a natural direction that we consider in this paper is to look for a temporal path that includes the maximum number of colors.

Related Works. Given a (static) vertex-colored graph, the problem of finding a colored path whose vertices have distinct colors and that includes the maximum number of colors (called tropical path) has been recently investigated in [8]. In [8], it is shown that the problem is not approximable, unless $P = NP$, within constant factor as the Longest Path problem, while hardness results or polynomial-time algorithms are given for several graph classes (bipartite chain graphs, threshold graphs, trees, block graphs, and proper interval graphs). A related problem on static graphs is that of finding a path whose vertices contain all the colors in a set and the vertices in the path are all colored distinctly [2, 16]. Another related problem considered in vertex-colored static graphs is that of finding the maximum number of vertex-disjoint uni-color paths [3, 11].

Several variants of the problem of finding a temporal path in a temporal vertex-colored graph that matches a given multiset of colors (called motif) have been introduced in [19]. It is shown in [19] that these variants of the problem are NP-complete, but fixed-parameter tractable when parameterized by the size of the motif.

Several problems related to finding paths in a temporal graph have been considered [21, 22]. A notable example is that of checking whether there exists a temporal path with waiting time constraint, a problem that has been shown to be NP-complete [5]. A similar problem is the temporal graph exploration [12] that asks for a temporal walk that, starting at a given vertex, visits all vertices of a graph with the smallest arrival time. Other related problems ask for the deletion of vertices so that temporal paths connecting pairs of vertices are removed [23]. Some recent contributions have investigated the computational complexity of exploring a temporal graph when the underlying graph is a star and finding an eulerian walk in a temporal graph [1, 4, 18].

Our Contribution. In this paper, given a temporal vertex-colored graph, we consider the problem of finding a temporal path whose vertices have distinct colors and that includes the maximum number of colors (a problem called Max CPTG). First, we study the approximation complexity of the Max CPTG problem and we show in Sect. 3 that it is not approximable within a factor $O(|V|^{\frac{1}{2}-\epsilon})$, unless $P = NP$. Notice that the corresponding problem on static graphs (finding a tropical path) is only known to be not approximable with a constant factor, unless $P = NP$ [8].

In Sect. 4 we present a heuristic for Max CPTG, as our aim is to design a method that is applicable even for a large number of colors. Notice that the

methods proposed in [19] are for different variants of the problem, where all the colors of the motif have to be included in a solution. Moreover, the methods proposed in [19] are fixed-parameter algorithms, where the parameter is the size of the motif, hence the running time of these latter algorithms is exponential in the size of the motif, leading to methods that are able to process motifs of moderate size (up to 18 colors are considered in [19]). On the other hand, we have to point out that the methods in [19] compute exact solutions, while our method is only a heuristic. In Sect. 5, we present an experimental evaluation of our heuristic, both on synthetic and real-world graphs. We start in Sect. 2 by introducing some definitions and by defining the problem we are interested in. Some proofs are omitted due to space constraints.

2 Preliminaries

We start this section by introducing the definition of discrete-time domain over which is defined a temporal graph.

Definition 1. *A discrete-time domain \mathcal{T} is a sequence of timestamp $t_i, 1 \leq i \leq t_{max}$, where each t_i is an integer and $t_i < t_{i+1}$. An interval $T = [t_i, t_j]$ over \mathcal{T} , where $t_i, t_j \in \mathcal{T}$ and $t_i \leq t_j$, is the sequence of timestamps t such that $t_i \leq t \leq t_j$.*

Two intervals $T_1 = [t_{a,1}, t_{b,1}]$ and $T_2 = [t_{a,2}, t_{b,2}]$ are disjoint if they do not share any timestamp, that is $t_{a,1} \leq t_{b,1} < t_{a,2} \leq t_{b,2}$ or $t_{a,2} \leq t_{b,2} < t_{a,1} \leq t_{b,1}$. The concatenation of T_1 and T_2 is an interval $T_1 \cdot T_2$ obtained by merging the two time intervals T_1 and T_2 , that is, assuming without loss of generality that $t_{a,1} \leq t_{b,1} < t_{a,2} \leq t_{b,2}$, $T_1 \cdot T_2 = [t_{a,1}, t_{b,2}]$. Given a set of pairwise disjoint intervals $T_1 = [t_{a,1}, t_{b,1}]$, $T_2 = [t_{a,2}, t_{b,2}]$, \dots , $T_q = [t_{a,q}, t_{b,q}]$, where $t_{a,1} \leq t_{b,1} < t_{a,2} \leq t_{b,2} < \dots < t_{a,q} \leq t_{b,q}$, we can define the concatenation of these intervals: $T_1 \cdot T_2 \cdot \dots \cdot T_q = [t_{a,1}, t_{q,2}]$.

We present now the definition of temporal graph. We assume that the vertex set is not changing on the time domain, that is the vertex set is identical in each timestamp.

Definition 2. *A temporal graph $G = (V, E, \mathcal{T})$ consists of*

1. *A set V of vertices*
2. *A time domain \mathcal{T}*
3. *A set $E \subseteq V \times V \times \mathcal{T}$ of temporal edges, where a temporal edge of G is a triple $\{u, v, t\}$, with $u, v \in V$ and $t \in \mathcal{T}$.*

$E[t]$ denotes the set of active edges at timestamp $t \in \mathcal{T}$, that is: $E[t] = \{\{u, v, t\} | \{u, v, t\} \in E\}$.

Now, we introduce the definition of temporal path.

Definition 3. *Given a temporal graph $G = (V, E, \mathcal{T})$, a temporal path in G is an alternating sequence of vertices and temporal edges $v_{p,1} e_{p,1} v_{p,2} e_{p,2} \dots e_{p,q-1} v_{p,q}$ such that:*

1. $v_{p,1}, v_{p,2}, \dots, v_{p,q}$ are distinct vertices
2. For each i , with $1 \leq i \leq q - 1$, $e_{p,i} = \{v_{p,i}, v_{p,i+1}, t_i\} \in E$, with $t_i \in \mathcal{T}$
3. For each i , with $1 \leq i \leq q - 1$, it holds $t_i < t_{i+1}$.

Vertices $v_{p,1}$ and $v_{p,q}$ in p are the start and end vertex of p . The length of p , denoted by $|p|$, is the number of vertices in p . We refer to Point 3 of Definition 3 as the *time constraint* of a temporal path.

A vertex-colored temporal graph is defined by adding a coloring to the vertices of a temporal graph.

Definition 4. $G_c = (V, E, \mathcal{T}, c)$ is a vertex-colored temporal graph, where $G = (V, E, \mathcal{T})$ is a temporal graph and $c : V \rightarrow C$ is a function that assigns a color from set C to each vertex in V .

We can now define the concept of colorful set of vertices.

Definition 5. Given a vertex-colored temporal graph $G_c = (V, E, \mathcal{T}, c)$, a set $V' \subseteq V$ is colorful if all the vertices in V' have distinct colors.

A temporal path in a vertex-colored temporal graph $G_c = (V, E, \mathcal{T}, c)$ is colorful if all its vertices have distinct colors. Now, we are able to define the problem we are interested into.

Problem 1. Maximum Colorful Path in a Temporal Graph (Max CPTG)

Input: A vertex-colored temporal colored graph $G = (V, \mathcal{T}, E, c)$.

Output: A colorful temporal path in G that includes the maximum number of colors (that is it has maximum length).

3 Inapproximability of Max CPTG

In this section we prove that the Max CPTG problem cannot be approximated within a factor $O(|V|^{\frac{1}{2}-\epsilon})$, unless $P = NP$. We prove this result by giving an approximation preserving reduction from the Maximum Independent Set problem (denoted by Max IS). For details on approximation preserving reductions see [20]. The Max IS problem, given a graph $G_I = (V_I, E_I)$, where $|V_I| = n$ and $|E_I| = m$, asks for an independent set $I \subseteq V_I$ of maximum size (we recall that I is an independent set if for $u, v \in I$, it holds that $\{u, v\} \notin E_I$).

Next, we describe our approximation preserving reduction from Max IS to Max CPTG. Given an instance $G_I = (V_I, E_I)$ of Max IS, we define a corresponding vertex-colored temporal graph $G = (V, \mathcal{T}, E, c)$, which is an instance of Max CPTG (an overview of $G = (V, \mathcal{T}, E, c)$ is given in Fig. 1).

For each $v_i \in V_I$, $1 \leq i \leq n$, V contains a set V_i of $n + 1$ vertices: $V_i = \{v_{i,x} : 0 \leq x \leq n\}$. Furthermore, V contains an additional set of vertices $V_{n+1} = \{v_{n+1,x} : 0 \leq x \leq n\}$. The vertex set V of G is defined as follows: $V = \bigcup_{i=1}^{n+1} V_i$.

The time domain \mathcal{T} consists of the concatenation of $n + 1$ time intervals $T(V_1), \dots, T(V_n), T(V_{n+1})$, where each $T(V_i)$, $1 \leq i \leq n + 1$, is associated with vertex set V_i . The idea is that only edges connecting vertices of V_i are active in

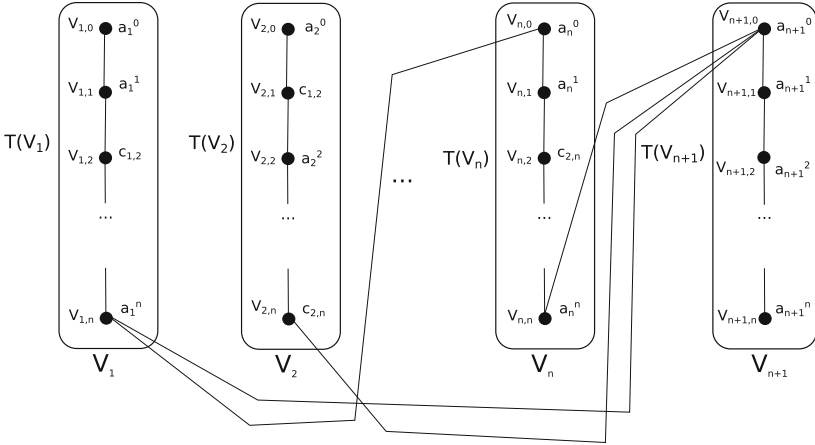


Fig. 1. An overview of the temporal graph $G = (V, T, E, c)$ associated with G_I . Each box contains the set V_i of vertices, $1 \leq i \leq n + 1$, and the path $p(V_i)$; the temporal edges are active in the time interval $T(V_i)$ (on the left of the box). Each vertex is labeled on the left with its name, on the right with its color. We assume that $\{v_1, v_2\} \in E_I$, hence $c(v_{1,2}) = c(v_{2,1}) = c_{1,2}$, $\{v_1, v_n\} \notin E_I$, hence $c(v_{1,n}) = a_1^n$, $c(v_{n,1}) = a_n^1$ and $\{v_2, v_n\} \in E_I$, hence $c(v_{2,n}) = c(v_{n,2}) = c_{2,n}$.

interval $T(V_i)$, except for the last timestamp. The interval $T(V_i)$, $1 \leq i \leq n + 1$, is defined as follows:

$$T(V_i) = [(n(i - 1) + i, (n + 1)i].$$

Notice, for example, that $T(V_1) = [1, n + 1]$ and $T(V_2) = [n + 2, 2n + 2]$ and so on. By construction, the intervals $T(V_i)$, $1 \leq i \leq n + 1$, are disjoint. The time domain \mathcal{T} is then the concatenation of intervals $T(V_1), T(V_2), \dots, T(V_{n+1})$, that is $\mathcal{T} = T(V_1) \cdot T(V_2) \cdot \dots \cdot T(V_n) \cdot T(V_{n+1})$.

The color function $c : V \rightarrow C$, is defined over the following set C of colors: $C = \{c_{i,0} : 1 \leq i \leq n + 1\} \cup \{c_{i,j} : \{v_i, v_j\} \in E_I \wedge i < j\} \cup \{a_i^q : 1 \leq i, q \leq n + 1\}$.

Essentially, each color $c_{i,j}$ encodes an edge $\{v_i, v_j\} \in E$, with $1 \leq i < j \leq n$, each color a_i^q , $1 \leq q \leq n$, encodes the fact that v_i is not adjacent to vertex v_q . Notice that $a_i^q \neq a_i^q$.

Now, we define the function c . For the vertices in V_i , with $1 \leq i \leq n$, c is defined as follows:

- $c(v_{i,0}) = a_i^0$
- $c(v_{i,x}) = c_{i,x}$, if $\{v_i, v_x\} \in E$ and $1 \leq i < x \leq n$
- $c(v_{i,x}) = c_{x,i}$, if $\{v_i, v_x\} \in E$ and $1 \leq x < i \leq n$
- $c(v_{i,x}) = a_i^x$, $1 \leq i, x \leq n$, if $\{v_i, v_x\} \notin E$

Notice that $c(v_{i,i}) = a_i^i$, for each i with $0 \leq i \leq n$, as we assume that G_I does not contain self loops.

For the vertices of V_{n+1} , the function c is defined as follows: $c(v_{n+1,x}) = a_{n+1,x}$, $0 \leq x \leq n$.

Next, we define the set of temporal edges of G . For each time interval $T(V_i)$, $1 \leq i \leq n + 1$, G contains a colorful temporal path $p(V_i)$ induced by the vertices $v_{i,x}$ with $0 \leq x \leq n$. The temporal edges active in interval $T(V_i)$ are defined as follows. At timestamp $t = n(i-1) + i + x$, with $0 \leq x \leq n-1$, $\{v_{i,x}, v_{i,x+1}, t\} \in E$; notice that $\{v_{i,x}, v_{i,x+1}, t\}$ is the only active temporal edge of G at timestamp t .

The temporal path $p(V_i)$ resulting from these edge is then:

$$p(V_i) = v_{i,0} \{v_{i,0}, v_{i,1}, n(i-1) + i\} v_{i,1} \{v_{i,1}, v_{i,2}, n(i-1) + i + 1\} \dots \\ \dots \{v_{i,n-1}, v_{i,n}, ni + i - 1\} v_{i,n}$$

Notice that, since by construction two intervals $T(V_i)$, $T(V_j)$, $1 \leq i < j \leq n$, are disjoint, the colorful temporal paths $p(V_i)$, $p(V_j)$ are active in disjoint intervals.

The set E contains also temporal edges defined to connect temporal colorful paths $p(V_i)$, $1 \leq i \leq n$. At timestamp $t = (n + 1)i$, $1 \leq i \leq n$, the following temporal edges belong to E :

- $\{v_{i,n}, v_{z,0}, t\}$, with $1 \leq i < z \leq n$, such that edge $\{v_i, v_z\} \notin E_I$
- $\{v_{i,n}, v_{n+1,0}, t\}$

This completes the definition of the vertex-colored temporal graph $G = (V, E, \mathcal{T}, c)$. We prove now a property of G .

Lemma 1. (\star) *Let $G_I = (V_I, E_I)$ be an instance of Max IS and let $G = (V, \mathcal{T}, E, c)$ be the corresponding instance of Max CPTG. Then:*

1. *Each temporal path $p(V_i)$, with $1 \leq i \leq n + 1$, is colorful*
2. *The vertices in temporal paths $p(V_i)$, $p(V_j)$, with $1 \leq i < j \leq n$ and $\{v_i, v_j\} \notin E_I$, have different colors.*

Now, we show how to construct in polynomial time a solution of Max CPTG from a solution of Max IS.

Lemma 2. *Let $G_I = (V_I, E_I)$ be an instance of Max IS and let $G = (V, \mathcal{T}, E, c)$ be the corresponding instance of Max CPTG. Given a solution $I \subseteq V_I$ of Max IS, we can construct in polynomial time a solution of Max CPTG of length at least $(|I| + 1)(n + 1)$.*

Proof. Consider an independent set $I = \{v_{i_1,1}, v_{i_2,2}, \dots, v_{i_b,b}\}$ of V_I , where $i_1 < i_2 < \dots < i_b$. Then define a solution p of Max CPTG as follows. The temporal path p includes the colored temporal paths $p(V_{i_x})$ in interval $T(V_{i_x})$, $1 \leq x \leq b$, and the temporal colored path $p(V_{n+1})$ in interval $T(V_{n+1})$. These colored paths are connected in p by the following temporal edges: $p(V_{i_x})$ and $p(V_{i_{x+1}})$, $1 \leq x \leq b - 1$ are connected by temporal edge $\{v_{i_x,n}, v_{i_{x+1},0}, t\}$ with $t = (n + 1)i_x$; $p(V_{i_b})$ and $p(V_{n+1})$ are connected by temporal edge $\{v_{i_b,n}, v_{n+1,0}, t\}$ with $t = (n + 1)i_b$.

Since $v_{i,x}, v_{i,y} \in V_I$, with $1 \leq x < y \leq b$, are not adjacent in G_I , it follows from Lemma 1 that the vertices in $p(V_{i_x})$ and $p(V_{i_y})$ do not share any color. Since each vertex in $p(V_{n+1})$ has a color distinct from the other vertices in V , it follows that p is colorful. Furthermore, notice that by construction, since the paths $p(V_i)$, $1 \leq i \leq n+1$, are defined over disjoint intervals, p is a temporal path. Finally, notice that p consists of $|I| + 1$ paths $p(V_i)$ each of length $n+1$, thus concluding the proof. \square

A solution of Max IS can be computed in polynomial time starting from a solution of Max CPTG.

Lemma 3. (\star) *Let $G_I = (V_I, E_I)$ be an instance of Max IS and let $G = (V, \mathcal{T}, E, c)$ be the corresponding instance of Max CPTG. Given a solution of Max CPTG of length $(q+1)(n+1)$, we can construct in polynomial time an independent set of G_I of size at least q .*

The inapproximability of Max CPTG follows from Lemma 2, Lemma 3 and from the inapproximability of Max IS [24].

Theorem 1. (\star) *Max CPTG is not approximable within a factor $O(|V|^{1/2-\varepsilon})$ unless $P = NP$.*

4 A Heuristic for Max CPTG

In this section, we present our efficient heuristic, called Colorful Temporal Path Local Search (CTPLS), for Max CPTG problem. CTPLS consists of two phases: (1) A greedy preliminary step that computes an initial solution, (2) A local search step that looks for a possible improvement of the solution.

We start by describing the preliminary greedy step. Given a vertex-colored temporal graph $G_c = (V, E, \mathcal{T}, c)$, first the step computes a segmentation of the time domain \mathcal{T} in $|C|$ disjoint intervals of equal length. Then it greedily looks for a temporal edge to be added to the path p computed so far in each interval. The path p is initialized as a temporal edge in the first interval. In the next intervals, the greedy step looks for a temporal edge that connects the last vertex of p to a vertex v that is not included in p .

Then CTPLS applies a local-search strategy, consisting of two different possible modifications of p (unless p contains all the colors).

1. **LS1** (Edge replacement): starting from the first edge of p , a temporal edge $\{u, v, t\}$ is possibly replaced with two temporal edges $\{u, x, t_1\}$, $\{x, v, t_2\}$, with $t_1 < t_2$; notice that vertex x is not already in p and it must be colored differently from the vertices already in p ; furthermore, all the temporal edges of the new path must satisfy the time constraint.
2. **LS2** (Vertex replacement): starting from the first vertex in the solution, it possibly replaces a vertex x in p and the temporal edges incident in x , with two vertices y and z and three temporal edges so that the new path satisfies the time constraint. Notice that y and z must not be in p and must have different colors from the vertices of p (except for the replaced vertex x).

5 Experimental Results

In this section, we present an experimental evaluation of CTPLS on synthetic and real networks. The CTPLS heuristic described in Sect. 4 is implemented in Python 3.7 using the NetworkX package for managing networks [14]. We perform the experiments on MacBook-Pro (OS version 11.4) with processor 2.9 GHz Intel Core i5 and 8 GB 2133 MHz LPDDR3 of RAM.

Synthetic Networks. In the first part of our experimental evaluation, we analyse the performance of CTPLS on synthetic datasets. We start by describing the synthetic datasets, then we discuss the results of CTPLS.

Datasets. Each synthetic graph is built as follows. First, we generate a temporal graph consisting of 500 vertices over 90 timestamps, such that the topology of the graph is based on one of the following models: Erdős-Renyi (ER) with parameter $p = 0.1$, Erdős-Renyi with parameter $p = 0.4$ and Barabasi-Albert (BA) with parameter equal to 10. $|C|$ vertices of the graph are then chosen randomly, assigned distinct colors and it is defined a temporal path that connects them. This ensures that each synthetic graph contains an optimal solution including all the colors in C , thus allowing to compare the solutions returned by CTPLS with an optimal one. Then each of the remaining vertices of the graph is assigned uniformly random colors from C . We consider the following sizes of C : 10, 20, 30 and 50 colors. For each graph model and for each size of C considered, we generated 20 independent synthetic graphs.

Outcome. We present in Table 1 the results of our experimental evaluation on the synthetic datasets. In particular, we report the minimum, maximum, average and standard deviation of the returned solutions of CTPLS over 20 instances for each color set and each graph model. Furthermore, we report the average running time (in seconds).

As reported in Table 1, the performances of CTPLS degrade with the increasing of the number of colors. For the BA-based graphs, for example, for a set of 10 colors the returned solutions contain on average at least 91% of the colors in C , for 50 colors the average number of colors contained in the returned solutions is 17.2 out of 50. The experimental results show also that the performances of CTPLS depend on the specific graph models. For the ER model with $p = 0.4$, the solutions returned are within 84% of the optimal solutions (for 50 colors). The performances are worse on ER with $p = 0.1$, within 28.6% of the optimal solution (for 50 colors). For the BA model, the solutions returned by CTPLS are close to the optimum only for the case of 10 colors (within 91% of the optimal solution) and are on average 83.75%, 47.5% and 34.4% for 20, 30 and 50 colors, respectively. It has to be pointed out that the Max CPTG problem is hard to approximate, as shown in Sect. 3, so it is not surprising that for some datasets the lengths of the solutions returned by CTPLS are not close to the optimum.

The method is always fast on synthetic datasets, requiring at most 0.68 seconds average running time (ER model with $p = 0.4$ and 50 colors).

Table 1. Performance of CTPLS on synthetic datasets, varying colors from 10 to 50. We report minimum, maximum, average and standard deviation over 20 independent synthetic networks for each different color set. The average running time is in seconds.

BA	Color 10		Color 20		Color 30		Color 50	
	Path	Time	Path	Time	Path	Time	Path	Time
Min	8	–	10	–	12	–	14	–
Max	10	–	16	–	17	–	21	–
Average	9.1	0.06	13.1	0.08	14.25	0.11	17.2	0.13
SD	0.79	–	1.41	–	1.65	–	2.28	–
ER $p = 0.1$	Color 10		Color 20		Color 30		Color 50	
	Path	Time	Path	Time	Path	Time	Path	Time
Min	9	–	11	–	9	–	5	–
Max	10	–	19	–	25	–	30	–
Average	9.85	0.09	16.75	0.15	18.45	0.14	14.3	0.11
SD	0.37	–	1.86	–	4.67	–	8.35	–
ER $p = 0.4$	Color 10		Color 20		Color 30		Color 50	
	Path	Time	Path	Time	Path	Time	Path	Time
Min	10	–	19	–	25	–	38	–
Max	10	–	20	–	30	–	46	–
Average	10	0.24	19.8	0.35	28.3	0.66	42.4	0.68
SD	0	–	0.41	–	1.17	–	1.82	–

Real Networks. In the second part of our experimental evaluation, we analyse the performance of CTPLS on four real-world datasets.

Datasets. We consider four different real-world temporal graphs taken from SNAP [17] for testing CTPLS¹: College messages (*CollegeMsg*), Email EU core (*email-Eu-core-temporal*), Bitcoin alpha (*soc-sign-bitcoinalpha*) and Bitcoin otc (*soc-sign-bitcoinotc*). These temporal graphs are not colored, hence, following the same approach of [19], we assigned uniformly random colors from a set of 30 colors and from a set of 50 colors. We consider two variants for each of this network, since the length of an optimal solution of Max CPTG on these graphs is unknown. Hence, in order to evaluate the results of CTPLS, for each real-world temporal graph we consider the original graph (denoted by NO-OP) and a modified temporal graph, called YES-OP, obtained by adding a temporal colorful path that contains each colors in C . This latter temporal graph contains an optimal solution of length $|C|$.

The first dataset, *CollegeMsg*, is taken from private messages sent on an online social network at the University of California, Irvine, where temporal edges represent private messages sent between users at a given time. The dataset

¹ <http://snap.stanford.edu/data/>.

contains 59835 temporal interactions, 1899 vertices and time domain \mathcal{T} of length $|\mathcal{T}| = 58911$. The *email-Eu-core-temporal* dataset is generated based on incoming and outgoing emails between members of a large European research institution, where temporal edges represent emails sent between users at a given time. This dataset contains 332334 temporal interactions, 986 vertices and time domain \mathcal{T} of length $|\mathcal{T}| = 207880$. *soc-sign-bitcoinalpha* and *soc-sign-bitcoinotc* are datasets of members who trade using Bitcoin on platforms called Bitcoin Alpha and Bitcoin OTC, respectively, to prevent transactions with risky users. A temporal edge $\{u, v, t\}$ represents a rate of member v given by member u at time t . *soc-sign-bitcoinalpha* contains 24186 temporal interactions, 3783 vertices and time domain \mathcal{T} of length $|\mathcal{T}| = 1647$, *soc-sign-bitcoinotc* contains 35592 temporal interactions, 5881 vertices and time domain \mathcal{T} of length $|\mathcal{T}| = 35445$.

Outcome. In Table 2 we report the number of colors included in the solutions returned by CTPLS and the running time (in minutes) for the two groups of real datasets we considered (NO-OP and YES-OP). As shown in Table 2, for the NO-OP networks with 30 colors, CTPLS found in the worst case a path containing 20 out of 30 colors (*soc-sign-bitcoinalpha*) and in the best case an optimal solution (*email-Eu-core-temporal*). For the other two networks, *CollegeMsg* and *soc-sign-bitcoinotc* networks, CTPLS found suboptimal solutions that contains a significative number of colors, 27 and 25 colors out of 30, respectively.

For the YES-OP networks with 30 colors, we don't report the result for *email-Eu-core-temporal*, as CTPLS was able to find an optimal solution for this dataset in NO-OP network. The results are not significantly different from the corresponding NO-OP datasets. CTPLS found in one case, the *CollegeMsg*, a path with the same number of colors as for the corresponding NO-OP network. In one case, (*soc-sign-bitcoinotc*) CTPLS found a larger number of colors (27 instead of 25 out of 30), in another case (*soc-sign-bitcoinalpha*) CTPLS found a slightly smaller number of colors (19 instead of 20 out of 30 colors). This decreasing is due to the fact that CTPLS considers a temporal edge that belongs to the YES-OP instance and not to the NO-OP instance and this prevents CTPLS to include all the vertices of the solution of the NO-OP instance.

For the NO-OP networks with 50 colors, CTPLS found in the worst case a path containing 36 colors (*soc-sign-bitcoinalpha*) and in the best case (*email-Eu-core-temporal*) 49 out of 50 colors. For the other two networks, *CollegeMsg* and *soc-sign-bitcoinotc* networks, CTPLS found 38 and 40 colors out of 50, respectively. For networks with 50 colors, CTPLS found the same number of colors in both YES-OP and NO-OP networks.

The experiments on real-world datasets confirm that CTPLS is able to produce suboptimal results even for networks with 50 colors. For the networks with 30 colors, CTPLS found solutions with at least 63% colors compared to the optimum (*soc-sign-bitcoinalpha*) and in one case an optimal solution. For the networks with larger number of colors (50 colors) CTPLS found solutions with at least 72% and at most 98% colors compared to the optimum. Except for (*soc-sign-bitcoinalpha*), the quality of solution returned by CTPLS starts slowly to degrade going from 30 colors to 50 colors. However, this deterioration is less pronounced than in synthetic datasets.

As for the running time, CTPLS is able to find a solution of Max CPTG in reasonable time, even for a set of 50 colors (notice that this value is larger than what has been considered in [19]). The running time varies considerably depending on the size of the temporal network and, in particular, on the length of the time domain. CTPLS indeed has highest running time on *CollegeMsg* and *email-Eu-core-temporal* whose time domain consists respectively of 207880 and 58911 timestamps. On the other hand, CTPLS requires at most 0.51 min on *soc-sign-bitcoinalpha* (NP-OP, 50 colors), which has the smallest time domain (1647 timestamps).

Table 2. Performance of CTPLS on real datasets. The value of the time (in minutes) and the value of return solution (path) for the Max CPTG problem is reported for two different color set (30 and 50).

NO-OP	Color 30		Color 50	
	Path	Time	Path	Time
CollegeMsg	27	144.71	38	27.91
email-Eu-core-temporal	30	52.60	49	129.05
soc-sign-bitcoinalpha	20	0.34	36	0.51
soc-sign-bitcoinotc	25	10.98	40	10.35
YES-OP	Color 30		Color 50	
	Path	Time	Path	Time
CollegeMsg	27	149.68	38	29.09
email-Eu-core-temporal	–	–	49	148.33
soc-sign-bitcoinalpha	19	0.16	36	0.22
soc-sign-bitcoinotc	27	6.34	40	9.82

6 Conclusion

In this paper, we have introduced a problem called Max CPTG for finding a colorful temporal path of maximum length in a vertex-colored temporal graph. We have studied the approximation complexity of the problem and we have provided an inapproximability lower bound. Then we have presented a heuristic (CTPLS) based on a greedy preliminary step and local search. We have provided an experimental evaluation, both on synthetic and real-world graphs.

Future works include the application of CTPLS to larger temporal networks. It would also be interesting to consider whether it is possible to apply the algebraic approach proposed in [19] to the Max CPTG problem and compare its performance with CTPLS.

References

1. Akrida, E.C., Mertzios, G.B., Spirakis, P.G., Raptopoulos, C.L.: The temporal explorer who returns to the base. *J. Comput. Syst. Sci.* **120**, 179–193 (2021)

2. Alon, N., Yuster, R., Zwick, U.: Color-coding. *J. ACM* **42**(4), 844–856 (1995)
3. Bonizzoni, P., Dondi, R., Pirola, Y.: Maximum disjoint paths on edge-colored graphs: approximability and tractability. *Algorithms* **6**(1), 1–11 (2013)
4. Bumpus, B.M., Meeks, K.: Edge exploration of temporal graphs. In: Flocchini, P., Moura, L. (eds.) *IWOCA 2021*. LNCS, vol. 12757, pp. 107–121. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-79987-8_8
5. Casteigts, A., Himmel, A., Molter, H., Zschoche, P.: Finding temporal paths under waiting time constraints. In: Cao, Y., Cheng, S., Li, M. (eds.) *In: Proceeding of the 31st International Symposium on Algorithms and Computation, ISAAC 2020*, pp. 30:1–30:18 (2020)
6. Castelli, M., Dondi, R., Hosseinzadeh, M.M.: Genetic algorithms for finding episodes in temporal networks. *Procedia Comput. Sci.* **176**, 215–224 (2020)
7. Choudhury, M.D., Feldman, M., Amer-Yahia, S., Golbandi, N., Lempel, R., Yu, C.: Automatic construction of travel itineraries using social breadcrumbs. In: Chignell, M.H., Toms, E.G. (eds.) *HT 2010, Proceedings of the 21st ACM Conference on Hypertext and Hypermedia 13–16 2010*, pp. 35–44 (2010)
8. Cohen, J., Italiano, G.F., Manoussakis, Y., Thang, N.K., Pham, H.P.: Tropical paths in vertex-colored graphs. *J. Comb. Optim.* **42**(3), 476–498 (2019). <https://doi.org/10.1007/s10878-019-00416-y>
9. Diestel, R.: *Graph Theory*. Graduate Texts in Mathematics, vol. 173, 4th edn. Springer, Heidelberg (2012)
10. Dondi, R., Hosseinzadeh, M.M.: Dense sub-networks discovery in temporal networks. *SN Comput. Sci.* **2**(3), 1–11 (2021)
11. Dondi, R., Sikora, F.: Finding disjoint paths on edge-colored graphs: more tractability results. *J. Comb. Optim.* **36**(4), 1315–1332 (2017). <https://doi.org/10.1007/s10878-017-0238-6>
12. Erlebach, T., Hoffmann, M., Kammer, F.: On temporal graph exploration. *J. Comput. Syst. Sci.* **119**, 1–18 (2021)
13. Gionis, A., Lappas, T., Pelechrinis, K., Terzi, E.: Customized tour recommendations in urban areas. In: Carterette, B., Diaz, F., Castillo, C., Metzler, D. (eds.) *Seventh ACM International Conference on Web Search and Data Mining, WSDM 2014*, pp. 313–322 (2014)
14. Hagberg, A., Swart, P., S Chult, D.: Exploring network structure, dynamics, and function using NetworkX. Technical report, Los Alamos National Lab. (LANL), Los Alamos, NM (United States) (2008)
15. Holme, P.: Modern temporal network theory: a colloquium. *Eur. Phys. J. B* **88**(9), 1–30 (2015). <https://doi.org/10.1140/epjb/e2015-60657-4>
16. Kowalik, L., Lauri, J.: On finding rainbow and colorful paths. *Theor. Comput. Sci.* **628**, 110–114 (2016)
17. Leskovec, J., Krevl, A.: SNAP Datasets: Stanford large network dataset collection, June 2014. <http://snap.stanford.edu/data>
18. Marino, A., Silva, A.: Königsberg sightseeing: Eulerian walks in temporal graphs. In: Flocchini, P., Moura, L. (eds.) *IWOCA 2021*. LNCS, vol. 12757, pp. 485–500. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-79987-8_34
19. Thejaswi, S., Gionis, A., Lauri, J.: Finding path motifs in large temporal graphs using algebraic fingerprints. *Big Data* **8**(5), 335–362 (2020)
20. Williamson, D.P., Shmoys, D.B.: *The Design of Approximation Algorithms*. Cambridge University Press, Cambridge (2011)
21. Wu, H., Cheng, J., Huang, S., Ke, Y., Lu, Y., Xu, Y.: Path problems in temporal graphs. *Proc. VLDB Endow.* **7**(9), 721–732 (2014)

22. Wu, H., Cheng, J., Ke, Y., Huang, S., Huang, Y., Wu, H.: Efficient algorithms for temporal path computation. *IEEE Trans. Knowl. Data Eng.* **28**(11), 2927–2942 (2016)
23. Zschoche, P., Fluschnik, T., Molter, H., Niedermeier, R.: The complexity of finding small separators in temporal graphs. *J. Comput. Syst. Sci.* **107**, 72–92 (2020)
24. Zuckerman, D.: Linear degree extractors and the inapproximability of max clique and chromatic number. In: Kleinberg, J.M. (ed.) *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, 21–23 2006, pp. 681–690 (2006)