A Fast Deterministic Detection of Small Pattern Graphs in Graphs without Large Cliques

Mirosław Kowaluk[1] and Andrzej Lingas[2]

[1] Institute of Informatics, University of Warsaw, Warsaw, Poland.
kowaluk@mimuw.edu.pl
[2] Department of Computer Science, Lund University
Andrzej.Lingas@cs.lth.se

Abstract. We show that for several pattern graphs on four vertices (e.g., C_4), their induced copies in host graphs with n vertices and no clique on $k+1$ vertices can be deterministically detected in time $\tilde{O}(n^{\omega}k^{\mu}+n^2k^2)$, where $\tilde{O}(f)$ stands for $O(f(\log f)^c)$ for some constant c, and $\mu \approx 0.46530$. The aforementioned pattern graphs have a pair of non-adjacent vertices whose neighborhoods are equal. By considering dual graphs, in the same asymptotic time, we can also detect four vertex pattern graphs, that have an adjacent pair of vertices with the same neighbors among the remaining vertices (e.g., K_4), in host graphs with n vertices and no independent set on $k+1$ vertices.
By using the concept of Ramsey numbers, we can extend our method for induced subgraph isomorphism to include larger pattern graphs having a set of independent vertices with the same neighborhood and n-vertex host graphs without cliques on $k+1$ vertices (as well as the pattern graphs and host graphs dual to the aforementioned ones, respectively).

Keywords: induced subgraph isomorphism, matrix multiplication, witnesses for Boolean matrix product, time complexity

1 Introduction

The problems of detecting subgraphs or induced subgraphs of a host graph that are isomorphic to a pattern graph are basic in graph algorithms. They are generally termed as *subgraph isomorphism* and *induced subgraph isomorphism* problems, respectively. Such well-known NP-hard problems as the independent set, clique, Hamiltonian cycle or Hamiltonian path can be regarded as their special cases.

Recent examples of applications of some variants of subgraph isomorphism include bio-molecular networks [1], social networks [19], and automatic design of processor systems [20]. In the aforementioned applications, the pattern graphs are typically of fixed size which allows for polynomial-time solutions.

For a pattern graph on k vertices and a host graph on n vertices, the fastest known general algorithms for subgraph isomorphism and induced subgraph isomorphism run in time $O(n^{\omega(\lfloor k/3\rfloor,\lceil (k-1)/3\rceil,\lceil k/3\rceil)})$ [4, 12, 17], where $\omega(p,q,r)$ denotes the exponent of fast matrix multiplication for rectangular matrices of size $n^p \times n^q$ and $n^q \times n^r$, respectively [15]. For $k \geq 6$, they also run in time $O(m^{\omega(\lfloor k/3\rfloor,\lceil (k-1)/3\rceil,\lceil k/3\rceil)/2})$ [4, 12], where m denotes the number of edges in a (connected) host graph. Further, we shall denote $\omega(1,1,1)$ by just ω. It is known that $\omega \leq 2.373$ [16, 23] and for example $\omega(1,2,1) \leq 3.257$ [15].

There are several known examples of pattern graphs of fixed size k for which one succeeded to design algorithms for subgraph isomorphism or/and induced subgraph isomorphism yielding asymptotic time upper bounds in terms of n lower than those offered by the aforementioned general method. For instance, for each pattern graph on k vertices having an independent set of size s, an isomorphic subgraph of an n-vertex graph can be (deterministically) detected in time $O(n^{\omega(\lceil (k-s)/2\rceil,1,\lfloor (k-s)/2\rfloor)}) \leq O(n^{k-s+1})$, assuming $k = O(1)$ [13]. Also, an induced subgraph isomorphic to the generalized diamond $K_k - e$, i.e., K_k with a single edge removed, as well as an induced subgraph isomorphic to the path on k vertices, P_k, can be detected in $O(n^{k-1})$ time [10, 21] which improves the general bound from [4] for $k \leq 5$.

More recent examples yields the randomized algorithm of Vassilevska Williams et al. for detecting an induced subgraph isomorphic to a pattern graph on four vertices different from K_4 and the four isolated vertices ($4K_1$) [22], subsuming similar randomized approach from [6]. Their algorithm runs in the same asymptotic time as that based on matrix multiplication for detecting triangles (i.e., K_3) from [11], i.e., in $O(n^\omega)$ time. The authors of [22] succeeded to obtain a deterministic version of their algorithm also running in the triangle asymptotic time for the diamond which is K_4 with one removed edge, denoted by $K_4 - e$. In fact there are few known earlier examples of pattern graphs on four vertices, different from the diamond, for which isomorphic induced subgraphs can be deterministically detected in an n vertex host graph substantially faster than by the general method. The earliest example is P_4, a path on four vertices, which can be detected in $O(n+m)$ time [2], where m is the number of edges in the host graph. The other example is a paw which is a triangle connected to the fourth vertex by an edge, denoted by $K_3 + e$. It can be detected in $O(n^\omega)$ time [18]. The third example is a claw which is a star with three leaves, it can be also detected in $O(n^\omega)$ time [4]. (Analogous

4-vertex pattern graph	deterministic time complexity
P_4	$O(n^2)$ [2]
claw	$O(n^{2.373})$ [4]
paw $K_3 + e$	$O(n^{2.373})$ [18]
diamond $K_4 - e$	$O(n^{2.373})$ [22]
C_4	$O(n^{3.257})$ [4]
K_4	$O(n^{3.257})$ [4]

Table 1. Known upper bounds on the deterministic time complexity of induced subgraph isomorphism for pattern graphs on four vertices.

upper bounds hold for the pattern graphs that are the complement to one of the aforementioned pattern graphs.)

Summarizing, there are twelve pairwise non-isomorphic pattern graphs on four vertices. For P_4 and its complement one can detect deterministically an isomorphic induced subgraph in an n vertex host graph in $O(n^2)$ time, while for the diamond, paw and claw, it takes $O(n^\omega)$ time. Thus, only for the cycle on four vertices, C_4, and its complement, and K_4 and its complement, there are no known deterministic algorithms for the induced subgraph isomorphism that are asymptotically faster than the general method yielding the $O(n^{\omega(1,2,1)})$-time bound. See also Table 1.

In this paper, we show in particular that if an n-vertex host graph does not contain a clique on $k+1$ vertices then for the pattern graph C_4 the induced subgraph isomorphism can be solved in time $\tilde{O}(n^\omega k^\mu + n^2 k^2)$, where $\tilde{O}(f)$ stands for $O(f(\log f)^c)$ for some constant c, and $\mu \approx 0.46530$. We also show that if the host graph does not contain a clique on $k+1$ vertices then one can detect an independent set on four vertices in the host graph in time $\tilde{O}(n^\omega k^\mu + n^2 k^2)$. Note that our upper time-bounds subsume the general $O(n^{\omega(1,2,1)}) \approx O(n^{3.257})$ bound for pattern graphs on four vertices for $k = O(n^{0.628})$. Our method works for all pattern graphs on four vertices that have a pair of non-adjacent vertices with the same neighborhood and all n-vertex host graphs without cliques on $k+1$ vertices in time $\tilde{O}(n^\omega k^\mu + n^2 k^2)$. By considering the dual graphs, we obtain the same asymptotic upper bound for the detection of an induced subgraph isomorphic to a given pattern graph with four vertices and a pair of adjacent vertices with the same neighbors among the remaining vertices in an n-vertex host graph without independent sets on $k+1$ vertices. (We denote the class of aforementioned 4-vertex pattern graphs and the class of pattern dual to them by $F_s^-(4)$ and $F_s^+(4)$, respectively.)

pattern graph class	deterministic time complexity	host graph with
$F_s^-(4)$, e.g., C_4 and $4K_1$	$\tilde{O}(n^\omega k^\mu + n^2 k^2)$	no $k+1$ clique
$F_s^+(4)$, e.g., K_4 and $2K_2$	$\tilde{O}(n^\omega k^\mu + n^2 k^2)$	no $k+1$ ind. set
$F_s^-(5)$, e.g., $K_{2,3}$, $(2,3)-fan$	$\tilde{O}(n^\omega (\frac{k(k+3)}{2})^\mu + n^2 (\frac{k(k+3)}{2})^3)$	no $k+1$ clique
$F_s^+(5)$, e.g., K_5 and $K_5 - e$	$\tilde{O}(n^\omega (\frac{k(k+3)}{2})^\mu + n^2 (\frac{k(k+3)}{2})^3)$	no $k+1$ ind. set

Table 2. Upper bounds on the deterministic time complexity of induced subgraph isomorphism for pattern graphs on four and five vertices and restricted host graphs presented in this paper ($\mu \approx 0.46530$). $K_{2,3}$ denotes the complete bipartite graph with two vertices on one side and three vertices on the other side. $(2,3)-fan$ can be obtained from $K_{2,3}$ by connecting the two vertex side by an edge.

By using the concept of Ramsey numbers, we can extend our method for induced subgraph isomorphism to include larger pattern graphs having a set of independent vertices with the same neighborhood and n-vertex host graphs without cliques on $k+1$ vertices (as well as the pattern graphs and host graphs dual to the aforementioned ones, respectively). In particular, we obtain an $\tilde{O}(n^\omega (\frac{k(k+3)}{2})^\mu + n^2 (\frac{k(k+3)}{2})^3)$ bound on the time complexity of induced subgraph isomorphism for pattern graphs with five vertices among which three are independent and have the same neighbors and n-vertex host graphs without cliques on $k+1$ vertices. (We denote the class of these 5-vertex pattern graphs by $F_s^-(5)$ and the class of graphs dual to them by $F_s^+(5)$, respectively.) See Table 2 for the summary of our results for pattern graphs on four and five vertices.

Our paper is structured as follows. In the next section, we provide basic definitions and facts. In Section 3, we present our algorithms for the detection of induced subgraphs isomorphic to pattern graphs on four vertices, in graphs without large cliques or large independent sets. In Section 4, we extend our algorithms for induced subgraph isomorphism to include larger pattern graphs. We conclude with Final Remarks.

2 Preliminaries

A *subgraph* of the graph $G = (V, E)$ is a graph $H = (V_H, E_H)$ such that $V_H \subseteq V$ and $E_H \subseteq E$.

An *induced subgraph* of the graph $G = (V, E)$ is a graph $H = (V_H, E_H)$ such that $V_H \subseteq V$ and $E_H = E \cap (V_H \times V_H)$.

The *neighborhood* of a vertex v in a graph G is the set of all vertices in G adjacent to v.

For $q \geq 4$, we shall distinguish the family $F_s^-(q)$ of pattern graphs H on q vertices $v_1, v_2, ..., v_q$ such that $v_1, ..., v_{q-2}$ form an independent set and have the same neighbors among the remaining two vertices $v_{q-1},\ v_q$. We shall also denote the family of pattern graphs dual to those in $F_s^-(q)$ by $F_s^+(q)$. Note that the latter family consists of pattern graphs H on q vertices $v_1, v_2, ..., v_q$ such that $v_1, ..., v_{q-2}$ form a clique and have the same neighbors among the remaining two vertices $v_{q-1},\ v_q$.

The *adjacency matrix* A of a graph $G = (V, E)$ is the $0 - 1$ $n \times n$ matrix where for $1 \leq i, j \leq n$, $A[i, j] = 1$ iff $\{i, j\} \in E$.

A *witness* for an entry $B[i, j]$ of the Boolean matrix product B of two Boolean matrices A_1 and A_2 is any index k such that $A_1[i, k]$ and $A_2[k, j]$ are equal to 1 [9].

Fact 1. *The fast matrix multiplication algorithm runs in $O(n^\omega)$ time, where ω is not greater than* 2.3728639 *[16] (cf. [23]).*

Fact 2. *The k-witness algorithm from [9] takes as input an integer k and two $n \times n$ Boolean matrices, and returns a list of q witnesses for each positive entry of the Boolean matrix product of those matrices, where q is the minimum of k and the total number of witnesses for this entry. It runs in $\tilde{O}(n^\omega k^{(3-\omega-\alpha)/(1-\alpha)} + n^2 k)$ time, where $\alpha \approx 0.30298$ (see [15]). One can rewrite the upper time bound as $\tilde{O}(n^\omega k^\mu + n^2 k)$, where $\mu \approx 0.46530$ [9].*

For two natural numbers $p,\ s$, the *Ramsey number* $R(p, s)$ is the minimum number ℓ such that any graph on at least ℓ vertices contains a complete subgraph (clique) on p vertices or an independent set on s vertices.

Fact 3. *For $p,\ s \geq 2$, $R(p, s) \leq \binom{p+s-2}{s-1}$ holds [3].*

3 The Algorithms

The idea of our algorithm for detecting induced subgraphs isomorphic to C_4 depicted in Fig. 1 is simple. First, we check if there is a pair of non-adjacent vertices connected by more than k different paths of length two in the host graph, by computing the arithmetic square of the adjacency matrix of the host graph. If so, there must be a pair of non-adjacent middle vertices of the aforementioned paths of length two, since all the middle vertices cannot induce a clique on more than k vertices. We conclude that the host graph contains a subgraph isomorphic to C_4. Otherwise, for each pair of non-adjacent vertices $i,\ j$, we examine the set of all middle vertices of paths of length two connecting i with j for the containment of

Input: a graph G on n vertices given by it adjacency matrix A and an integer parameter $k \in [2, n]$ such that G does not contain a $(k+1)$-clique.

Output: if G contains an induced subgraph isomorphic to C_4 then YES else NO

```
C ← A × A (arithmetic product)
for i = 1 to n do
    for j = 1 to n do
        if i ≠ j ∧ A[i, j] = 0 ∧ C[i, j] > k then
        return YES and stop
    end for
end for
for i = 1 to n do
    for j = 1 to n do
        if i ≠ j ∧ A[i, j] = 0 then W[i, j] ← the set of witnesses for the B[i, j] entry
        of the Boolean product B of A with A
        if W[i, j] contains at least two vertices and it does not induce a clique in G
        then
        return YES and stop
    end for
end for
return NO
```

Fig. 1. An algorithm for detecting an induced subgraph isomorphic to C_4 in a graph with no $(k+1)$-clique.

a pair of non-adjacent vertices. Since the aforementioned middle vertices correspond to witnesses for the (i, j) entry of the Boolean square of the adjacency matrix of the host graph, we can use the k-witness algorithm from [9] described in Fact 2 to compute the middle vertices.

Lemma 1. *Algorithm 1 is correct.*

Proof. Let B be the Boolean square of the adjacency matrix A. Witnesses of $B[i, j]$ are just middle vertices of different paths of length two connecting the vertices i and j. The answer YES is returned if $A[i, j] = 0$ and the set of witnesses contains at least two vertices and it does not form a clique. Then, i and j are not adjacent, and there are two paths of length two connecting i and j whose middle vertices are not adjacent. Thus, there is an induced C_4 in G.

To prove the correctness of the answer NO suppose that there is an induced C_4 in G. We may assume w.l.o.g. that it is induced by the vertices $q,\ r,\ i,\ j$, where $A[i, j] = 0$ and $A[q, r] = 0$. Note that q and r belong to the set $W[i, j]$ of witnesses of $B[i, j]$. Since $A[q, r] = 0$, $W[i, j]$ does not induce a clique. Consequently, Algorithm 1 would return YES. □

Lemma 2. *Algorithm 1 runs in time $O(n^\omega + T(n,k) + n^2k^2)$, where $T(n,k)$ stands for the time necessary to solve the k-witness problem for two $n \times n$ Boolean matrices.*

Proof. The computation of C takes $O(n^\omega)$ time. The first double loop takes $O(n^2)$ time. To implement the second double loop we need to solve the k-witness problem. It takes $T(n,k)$ time. After that, the second double loop takes $O(n^2k^2)$ since the considered sets $W[i,j]$ are of size not exceeding k, and we can test if $l \leq k$ vertices induce a clique in $O(k^2)$ time. □

By combining Lemmata 1, 2 with Fact 2, we obtain our first main result.

Theorem 1. *Let G be a graph on n vertices with no clique on $k+1$ vertices. We can decide if G contains an induced subgraph isomorphic to C_4 in time $\tilde{O}(n^\omega k^\mu + n^2k^2)$, where $\mu \approx 0.46530$.*

Recall that $F_s^-(4)$ is the family of graphs H on four vertices v_1, v_2, v_3, v_4 such that $v_1,\ v_2$ are not adjacent and have the same neighbors among the remaining two vertices $v_3,\ v_4$. Clearly, C_4 belongs to $F_s^-(4)$. Also, $4K_1$ (i.e, an independent set on four vertices), $K_4 - e$, $P_3 + K_1$ and the paw belong to $F_s^-(4)$.

We can immediately generalize Algorithm 1 and Theorem 1 to include the detection of a given member $H \in F_s^-(4)$ in the graph G satisfying the requirements of Theorem 1.

Consider the distinguished pair of non-adjacent vertices $v_1,\ v_2$ in H. Let $v_3,\ v_4$ be the two remaining vertices. Finally, let $\bar{H}$ stand for the graph dual to H, and $\bar{A}$ for the adjacency matrix of $\bar{H}$. In the generalized Algorithm 1, we match $v_3,\ v_4$ with $i,\ j$, respectively, and $v_1,\ v_2$, with witnesses for the entry corresponding to (i,j) of an appropriated Boolean matrix product. In case of C_4, both v_3 and v_4 are neighbors of v_1 and v_2, so we use the Boolean product of A with A. Generally, for $H \in F_s^-(4)$, we have to replace the Boolean product $A \times A$ with that of two matrices $A_1,\ A_2$ defined as follows:

- if $\{v_3, v_1\}$, $\{v_3, v_2\}$ are edges of H then $A_1 = A$,
- otherwise, $\{v_3, v_q\}$, $\{v_3, v_2\}$ are edges of $\bar{H}$ and $A_1 = \bar{A}$,
- if $\{v_1, v_4\}$, $\{v_2, v_4\}$ are edges of H then $A_2 = A$,
- otherwise, $\{v_1, v_4\}$, $\{v_2, v_4\}$ are edges of $\bar{H}$ and $A_2 = \bar{A}$.

Also, to count the number of witnesses for respective entries of the Boolean product of A_1 and A_2, in the first step of the generalized algorithm, we compute the arithmetic product of A_1 and A_2 treated as

Input: a graph G on n vertices given by it adjacency matrix A, a pattern graph H on vertices v_1, v_2, v_3, v_4 satisfying the requirements of the membership in $F_s^-(4)$, and an integer parameter $k \in [2, n]$ such that G does not contain a $(k+1)$-clique.
Output: if G contains an induced subgraph isomorphic to H then YES else NO

```
Ā ← the adjacency matrix of the complement of G
if {v3, v1} is an edge of H then A1 ← A else A1 ← Ā
if {v1, v4} is an edge of H then A2 ← A else A2 ← Ā
C ← A1 × A2 (arithmetic product)
if {v3, v4} is an edge of H then a ← 1 else a ← 0
for i = 1 to n do
  for j = 1 to n do
    if i ≠ j ∧ A[i, j] = a ∧ C[i, j] > k then
    return YES and stop
  end for
end for
for i = 1 to n do
  for j = 1 to n do
    if i ≠ j ∧ A[i, j] = a then W[i, j] ← the set of witnesses for the B[i, j] entry
    of the Boolean product B of A1 with A2
    if W[i, j] contains at least two vertices and it does not induce a clique in G
    then
    return YES and stop
  end for
end for
return NO
```

Fig. 2. An algorithm for detecting an induced subgraph isomorphic to $H \in F_s^-(4)$ in a graph with no $(k+1)$-clique.

arithmetic matrices, instead of the arithmetic product of A with A. Finally, if $\{v_3,\ v_4\}$ is an edge of H then we have to replace the condition $A[i,j] = 0$ with $A[i,j] = 1$.

By arguing analogously as in the proof of Theorem 1, we obtain the following generalization.

Theorem 2. *Let $H \in F_s^-(4)$, and let G be a graph on n vertices with no clique on $k+1$ vertices. We can decide if G contains an induced subgraph isomorphic to H in time $\tilde{O}(n^{\omega}k^{\mu} + n^2k^2)$, where $\mu \approx 0.46530$.*

Recall that $F_s^+(4)$ is the family of graphs on four vertices dual to those in $F_s^-(4)$, i.e., graphs that have a pair of adjacent vertices with the same neighborhood. Note that in particular K_4, $K_3 + K_1$, $2K_2$, and again $K_4 - e$ belong to $F_s^+(4)$.

By considering dual graphs, we obtain the following corollary from Theorem 2.

Corollary 1. *Let $H \in F_s^+(4)$, and let G be a graph with n vertices and no independent set on $k+1$ vertices. We can decide if G contains an induced subgraph isomorphic to H in time $\tilde{O}(n^\omega k^\mu + n^2 k^2)$, where $\mu \approx 0.46530$.*

Corollary 2. *Let $H \in F_s^-(4) \cap F_s^+(4)$, and let G be a graph on n vertices which does not contain a clique on $k+1$ vertices or an independent set on $k+1$ vertices. We can decide if G contains an induced subgraph isomorphic to H in time $\tilde{O}(n^\omega k^\mu + n^2 k^2)$, where $\mu \approx 0.46530$.*

Note that $F_s^-(4) \cap F_s^+(4) = \{K_4 - e,\ K_2 + 2K_1\}$.

4 Extensions to Larger Pattern Graphs

Recall that for $q \geq 4$, $F_s^-(q)$ stands for the family of graphs H on q vertices $v_1, v_2, ..., v_q$ such that $v_1, ..., v_{q-2}$ form an independent set and have the same neighbors among the remaining two vertices $v_{q-1},\ v_q$.

We can easily generalize Algorithm 2 and Theorem 2 to include the detection of a given member $H \in F_s^-(q)$, where $q \geq 4$, in the graph G satisfying the requirements of Theorem 2.

In Algorithm 2, it is sufficient for a pair of vertices $i,\ j$ satisfying $A[i,j] = a$ to deduce or verify that the set of witnesses for the (i,j) entry of the Boolean matrix product contains a pair of non-adjacent vertices in order to detect an induced subgraph isomorphic to the pattern graph H. Now, we have to deduce or verify that the aforementioned set of witnesses contains an independent set on $q-2$ vertices in G instead. If the set of witnesses contains no less than the Ramsey number $R(k+1, q-2)$ of vertices then it fulfills this requirement since we assume that the host graph G does not contain a clique on $k+1$ vertices. Otherwise, we have to go through all subsets of $q-2$ witnesses to check if any of them forms an independent set. If we do not know the exact Ramsey number $R(k+1, q-2)$ but only an upper bound t on $R(k+1, q-2)$, we have to use t instead. Thus the threshold on the number of witness becomes now $t-1$ instead of k, and we have to use Fact 2 to find up to $t-1$ witnesses. The generalized algorithm is depicted in Fig. 3.

Lemma 3. *Algorithm 3 runs in time $\tilde{O}(n^\omega (t-1)^\mu + n^2 (t-1)^{q-2})$, where $\mu \approx 0.46530$.*

Proof. As a straightforward generalization of Algorithm 2, Algorithm 3 runs in time $O(n^\omega + T(n,t) + n^2 t^{q-2})$, where $T(n,t)$ stands for the time necessary to solve the $(t-1)$-witness problem for two $n \times n$ Boolean matrices. By Fact 2, we obtain the lemma. □

Input: a graph G on n vertices given by it adjacency matrix A, a pattern graph H on $q \geq 4$ vertices $v_1, ..., v_q$ satisfying the requirements of the membership in $F_s^-(q)$, an integer parameter $k \in [2, n]$ such that G does not contain a $(k+1)$-clique, and an upper bound t on the Ramsey number $R(k+1, q-2)$.

Output: if G contains an induced subgraph isomorphic to H then YES else NO

$\bar{A} \leftarrow$ the adjacency matrix of the complement of G
if $\{v_q, v_1\}$ is an edge of H **then** $A_1 \leftarrow A$ **else** $A_1 \leftarrow \bar{A}$
if $\{v_1, v_{q-1}\}$ is an edge of H **then** $A_2 \leftarrow A$ **else** $A_2 \leftarrow \bar{A}$
$C \leftarrow A_1 \times A_2$ (arithmetic product)
if $\{v_{q-1}, v_q\}$ is an edge of H **then** $a \leftarrow 1$ **else** $a \leftarrow 0$
for $i = 1$ **to** n **do**
    for $j = 1$ **to** n **do**
        if $i \neq j \wedge A[i,j] = a \wedge C[i,j] \geq t$ **then**
        return YES and stop
    end for
end for
for $i = 1$ **to** n **do**
    for $j = 1$ **to** n **do**
        if $i \neq j \wedge A[i,j] = a$ **then** $W[i,j] \leftarrow$ the set of witnesses for the $B[i,j]$ entry of the Boolean product of A_1 with A_2
        if $W[i,j]$ contains an independent set on $q-2$ vertices in G **then**
        return YES and stop
    end for
end for
return NO

Fig. 3. An algorithm for detecting an induced subgraph isomorphic to $H \in F_s^-(q)$, $q \geq 4$, in a graph with no $(k+1)$-clique.

Theorem 3. *Let $q \geq 4$, $H \in F_s^-(q)$, and let G be a graph on n vertices with no clique on $k+1$ vertices. Next, let t be a known upper bound on $R(k+1, q-2)$. We can decide if G contains an induced subgraph isomorphic to H in time $\tilde{O}(n^\omega (t-1)^\mu + n^2 (t-1)^{q-2})$, where $\mu \approx 0.46530$.*

Proof. We use Algorithm 3. Its correctness follows from the discussion preceding its pseudocode. Lemma 3 yields the time bound. □

By Fact 3, we can easily conclude that $R(k+1, 3) \leq \frac{(k+2)(k+1)}{2}$. Hence, we obtain the following corollary.

Corollary 3. *Let $H \in F_s^-(5)$, and let G be a graph on n vertices with no clique on $k+1$ vertices. We can decide if G contains an induced subgraph isomorphic to H in time $\tilde{O}(n^\omega (\frac{k(k+3)}{2})^\mu + n^2 (\frac{k(k+3)}{2})^3)$.*

Note that $H \in F_s^-(5)$ includes $K_{2,3}$, $(2,3) - fan$ (see Table 2), $K_2 + 3K_1$ and $5K_1$ among other things. We leave to the reader stating the results implied by Theorem 3 and Corollary 3 for the dual pattern and host graphs.

5 Final Remarks

The authors of [12] have shown that if one knows the number of induced subgraphs of an n vertex host graph that are isomorphic to a given 4-vertex pattern then one can compute the analogous number for each of the twelve 4-vertex pattern graphs in $O(n^{\omega})$ time. This result has been generalized to include pattern graphs on more than four vertices in [13]. Thus, in particular, if we knew that the host graph is free from K_4 then we could compute for each pattern graph H on four vertices the number of induced subgraphs isomorphic to H in $O(n^{\omega})$ time. Generally, graphs with some forbidden subgraphs, induced subgraphs, or minors (e.g., planar graphs) are widely studied in algorithmics. Typically, the forbidden subgraphs are of small fixed size. In our approach the forbidden $(k+1)$ clique (or, an $(k+1)$ independent set, respectively) can be very large, e.g., even larger than $\sqrt{n}$ and we can still obtain an upper time-bound on detecting for instance induced subgraphs isomorphic to C_4 better than the known $O(n^{\omega(1,2,1})$ one.

One of the reviewers posed an interesting question of whether or not our pattern detection algorithms can be extended to include pattern finding algorithms without substantially increasing their running times. For instance, consider our algorithm for C_4 detection. Suppose that i, j is a pair of non-adjacent vertices in the input graph. If C_4 is detected by finding a pair of non-adjacent witnesses among at most k witnesses of $B[i,j]$, then we can easily locate an induced subgraph isomorphic to C_4. However, if C_4 is detected by checking that $B[i,j]$ has more than k witnesses then we need to know at least $k+1$ witnesses of $B[i,j]$ in order to locate a pair of non-adjacent ones. Thus, we can extend our algorithm for C_4 and the other ones to the finding variant by increasing the number of witnesses to compute for each positive entry of B by one. This increases the running times of our algorithms solely marginally.

Acknowledgments The research has been supported in part by the grant of polish National Science Center 2014/13/B/ST6/00770 and Swedish Research Council grant 621-2011-6179, respectively.

References

1. Alon, N., Dao, P., Hajirasouliha, I., Hormozdiari, F., Sahinalp, S. C.: Biomolecular network motif counting and discovery by color coding. Bioinformatics (ISMB 2008),

24(13) (2008), pp. 241–249
2. Corneil, D.G., Perl, Y., Stewart, L.K.: A Linear Recognition Algorithm for Cographs. SIAM J. Comput. 14(4), pp. 926–934 (1985)
3. Chung, F.R.K., Grinstead, C.M.: A Survey of Bounds for Classical Ramsey Numbers. Journal of Graph Theory, vol. 7 (1983) pp. 25–37
4. Eisenbrand, F., Grandoni, F.: On the complexity of fixed parameter clique and dominating set. Theoretical Computer Science 326, pp. 57–67 (2004)
5. Eschen, E.M., Hoàng, C.T., Spinrad, J., Sritharan, R.: On graphs without a C4 or a diamond. Discrete Applied Mathematics 159(7), pp. 581–587 (2011)
6. Floderus, P., Kowaluk, M., Lingas, A., Lundell, E.-M.: Detecting and Counting Small Pattern Graphs. SIAM J. Discrete Math. 29(3), pp. 1322–1339 (2015)
7. Floderus, P., Kowaluk, M., Lingas,Ã., Lundell, E.-M.: Induced subgraph isomorphism: Are some patterns substantially easier than others?. In: Proc. COCOON 2012, LNCS 7434, pp. 37–48 (2012)
8. Garey, M.R., Johnson, D.S.: Computers and Intractability - A Guide to the Theory of NP-Completeness. Bell Laboratories, 1979, Murray hill, New Jersey
9. Gąsieniec, L., Kowaluk, M., Lingas, A.: Faster multi-witnesses for Boolean matrix product. Information Processing Letters 109, pp. 242–247 (2009)
10. Hoàng, C.T., Kaminski, M., Sawada,. J., Sritharan, R.: Finding and listing induced paths and cycles. Discrete Applied Mathematics 161(4-5), pp. 633–641 (2013)
11. Itai, A., Rodeh, M..: Finding a minimum circuit in a graph. SIAM Journal of Computing, vol. 7, 1978
12. Kloks, T., Kratsch, D., Müller, H.: Finding and counting small induced subgraphs efficiently. Information Processing Letters 74(3-4), pp. 115–121 (2000)
13. Kowaluk, M., Lingas, A., Lundell, E.-M.: Counting and detecting small subgraphs via equations and matrix multiplication. SIAM J. of Discrete Mathematics 27(2), pp. 892–909 (2013)
14. Kuramochi, M., Karypis, G.: Finding Frequent Patterns in a Large Sparse Graph. Data Mining and Knowledge Discovery, 11, pp. 243–271, 2005
15. Le Gall, F.: Faster Algorithms for Rectangular Matrix Multiplication. In: Proc. 53rd Symposium on Foundations of Computer Science (FOCS), pp. 514–523 (2012)
16. Le Gall, F.: Powers of Tensors and Fast Matrix Multiplication. In: Proc. 39th International Symposium on Symbolic and Algebraic Computation, pp. 296–303 (2014)
17. Nešetřil, J., Poljak, S.: On the complexity of the subgraph problem. Commentationes Mathematicae Universitatis Carolinae, 26(2), 1985, pp. 415–419
18. Olariu, S.: Paw-Free Graphs. Information Processing Letters 28, pp. 53–54, (1988)
19. Schank, T., Wagner, D.: Finding, Counting and Listing All Triangles in Large Graphs, an Experimental Study. In: Proc. WEA 2005, pp. 606–609
20. Wolinski, C., Kuchcinski, K., Raffin, E.: Automatic Design of Application-Specific Reconfigurable Processor Extensions with UPaK Synthesis Kernel. ACM Transactions on Design Automation of Electronic Systems, 15(1) (2009), pp. 1–36
21. Vassilevska, V.: Efficient Algorithms for Path Problems in Weighted Graphs. PhD thesis, CMU, CMU-CS-08-147, 2008
22. Vassilevska Williams, V., Wang, J.R., Williams, R., Yu H.: Finding Four-Node Subgraphs in Triangle Time. In Proc. of SODA 2015, pp. 1671–1680
23. Vassilevska Williams, V.: Multiplying matrices faster than Coppersmith-Winograd. In: Proc. 44th Annual ACM Symposium on Theory of Computing (STOC), pp. 887–898 (2012)