On the complexity of hamiltonian path and cycle problems in certain classes of digraphs

Jørgen Bang-Jensen * Gregory Gutin †

Abstract

We survey results on the sequential and parallel complexity of hamiltonian path and cycle problems in various classes of digraphs which generalize tournaments. We give detailed informations on the difference in difficulties for these problems for the various classes as well as prove new results on hamiltonian paths starting in a specified vertex for a quite general class of digraphs.

1 Introduction

This paper is based in part on a invited plenary talk given by the first author at ODSA '97 in Rostock, September 8-10 1997. The purpose of the paper is to survey results on the complexity of hamiltonian path and cycle problems in generalizations of tournaments and point out similarities and differences among the various classes. There have been recent surveys by the authors on tournaments and generalizations of tournaments, respectively [10, 11], but contrary to those papers, in this paper we focus explicitly on the complexity on hamiltonian path and cycle problems and give a number of quite detailed explanations which we hope will inspire many readers to explore this rich area by themselves and in any case will give the readers a feeling for the techniques used in this area. As another difference to [10, 11] this current survey contains a number of results on the parallel complexity of hamiltonian path and cycle problems for generalizations of tournaments and finally we include proofs of new results on hamiltonian paths starting or ending at a specified vertex in a quite general class of digraphs.

It is well known that the hamiltonian path and cycle problems for general digraphs as well as their numerous modifications are $\mathcal{NP}$-complete. Hence, it makes sense to investigate classes of digraphs where the hamiltonian path and cycle problems can be solved in polynomial time. A well known example of such class is tournaments. In this paper, we describe algorithmic results obtained for this class of digraphs as well as for wider classes of digraphs which generalize tournaments, including the rather wide

*Department of Mathematics and Computer Science, Odense University, DK-5230, Denmark email jbj@imada.ou.dk fax : 65932691

†Department of Mathematics and Statistics, Brunel University, Uxbridge, Middlesex UP8 3PH United Kingdom, email z.g.gutin@brunel.ac.uk

classes of totally Φ_i-decomposable digraphs ($i = 0, 1$) defined in Section 6. Moreover, we state some challenging open problems and conjectures.

We consider the following problems: Given a digraph D and two vertices x, y in D, decide if there exists

1. a hamiltonian path of D (the hamiltonian path problem, or the **HP** problem, for short);
2. a hamiltonian cycle of D (the **HC** problem);
3. a hamiltonian path of D starting at x (the **HPx** problem);
4. a hamiltonian path of D between x and y (the order of x and y is not specified) (the **HP**[**x,y**] problem);
5. a hamiltonian path of D from x to y (the **HPxy** problem).

For many of the cases we consider, one can construct polynomial algorithms using complete theoretical characterizations and their constructive proofs. In other, more difficult cases, polynomial algorithms are obtained by using either partial theoretical results (the **HPxy** problem for tournaments, for example) or by transforming a problem into ones having theoretical characterizations (for example the **HP** and **HC** problems for totally Φ_i-decomposable digraphs).

2 Terminology

For terminology on parallel algorithms we refer the reader to [40]. The class $\mathcal{NC}$ is the class of problems for which there exits a parallel algorithm solving the problem in polylogarithmic time and using a polynomial number of processors (both wrt the size of the problem). The graph-theoretical terminology is fairly standard, generally following [25]. We shall always use the number n to denote the number of vertices in the digraph currently under consideration. Digraphs are finite, have no loops or multiple arcs. $V(D)$ and $A(D)$ denote the vertex set and the arc set of a digraph D. The number of vertices in a digraph is its *order*. We shall denote the arc from a vertex x to a vertex y by xy. If $xy \in A(D)$, we shall say that x dominates y and denote it by $x \rightarrow y$. The *out-neighbourhood* (*in-neighbourhood*) of a vertex x in a digraph D is the set of all vertices of D dominated by (dominating) x. We shall denote the *in-neighbourhood and out-neighbourhood* of vertex x by $N^-(x)$ and $N^+(x)$, respectively. For disjoint subsets $H, K \subset V(D)$ we use the notation $H \Rightarrow K$ to denote that there are no arcs from K to H.

By a *cycle (path,* respectively) we mean a directed (simple) cycle (path, respectively). If C is a cycle and x is a vertex on C, then we denote by x^- (respectively, x^+) the predecessor (respectively, the successor) of x on C. Sometimes we shall use this notation for vertices on different cycles, but the meaning should always be clear. If R is a cycle or a path with two vertices u, v such that u can reach v on R, then $R[u, v]$ denotes the subpath of R from u to v. A cycle (path) of a digraph D is *hamiltonian* if it contains all the vertices of D. A digraph is *hamiltonian* if it has a hamiltonian cycle.

An (x,y)-*path* is a path from x to y. A digraph D is *strongly connected* (or just *strong*) if there exists an (x,y)-path and a (y,x)-path for every choice of distinct vertices x,y of D. The digraph D is k-*strongly-connected* if $|V(D)| \geq k+1$ and $D-X$ is strongly connected for any $X \subset V(D)$ with $|X| \leq k-1$.

The *underlying graph* of a digraph D is the graph obtained from D by disregarding the orientations of all arcs of D. We shall denote the underlying graph of D by $U[D]$ and say that D is *connected* if $U[D]$ is connected. A digraph T is *semicomplete* if $U[T]$ is complete. A *tournament* is a semicomplete digraph with no cycles of length 2.

A collection F of pairwise vertex disjoint paths and cycles of a digraph D is called a k-*path-cycle factor* of D if F covers $V(D)$ and has exactly $k \geq 0$ paths. F is called a k-*path factor* if it is contains only paths. We shall call a 0-path-cycle factor a *cycle factor*.

An *out-branching (respectively, in-branching)* rooted at some vertex v in a digraph D is a spanning tree in $U(D)$ which is oriented (in D) in such a way that every vertex $x \neq v$ has precisely one arc coming in (respectively, going out).

3 Various classes of generalizations of tournaments

A digraph D is *locally in-semicomplete* (*locally out-semicomplete*, respectively) if, for every vertex x of D, the in-neighbourhood of x (its out-neighbourhood, respectively) induces a semicomplete digraph. A digraph D is *locally semicomplete* if it is both locally in- and out-semicomplete. We shall use the abbreviation LSD's (LISD's and LOSD's, respectively) for locally semicomplete digraphs (locally in-semicomplete and out-semicomplete digraphs, respectively). A digraph D is called a *semicomplete k-partite digraph* ($k \geq 2$) or a *semicomplete multipartite digraph* (abbreviated to SMD) if $U[D]$ is a complete k-partite graph. The special case when $k=2$ is called a *semicomplete bipartite digraph* (abbreviated to SBD). If D is a semicomplete k-partite digraph we call the maximal independent sets of D the *colour classes* of D and denote these $V_1,\ldots,V_k$.

A digraph D is *path-mergeable* if, for every choice of vertices $x,y \in V(D)$ and every pair of internally disjoint (x,y)-paths P,P', there exists an (x,y)-path P^* in D, such that $V(P^*) = V(P) \cup V(P')$.

A digraph D is called *quasi-transitive* if, for every triple x,y,z of distinct vertices of D such that xy and yz are arcs of D, there is at least one arc between x and z. Quasi-transitive digraphs were introduced by A. Ghouilá-Houri [29] who proved that the underlying graphs of quasi-transitive digraphs are precisely comparability graphs.

Let D be a digraph on p vertices $v_1,...,v_p$ and let $L_1,...,L_p$ be a disjoint collection of digraphs. Then $D[L_1,...,L_p]$ is the new digraph obtained from D by replacing each vertex v_i of D by L_i and adding an arc from every vertex of L_i to every vertex of L_j if and only if (v_i,v_j) is an arc of D ($1 \leq i \neq j \leq p$). Let Φ be a set of digraphs, containing the digraph on one vertex and no arcs. A $\Phi-$*graph* is a member $D \in \Phi$. A digraph D is an *extended* $\Phi-$graph if either it has only one vertex, or there is a decomposition $D = R[H_1,...,H_r]$ such that $R \in \Phi$, each of the digraphs $H_i,\ i=1,...,r$ has no arcs, and $r \geq 2$. A set Φ *is closed with respect to extension* if each extended Φ-graph is in Φ. In particular, SMD's form such a closed set. The set of semicomplete digraphs is not closed in this sense. Extended digraphs appear in the solutions of some problems, especially, in the study of different sets of totally Φ-decomposable digraphs (see [36] and Section 6). We describe several results on extended semicomplete digraphs in Section 5.

4 Tournaments

The problems **HP,HC, HPx, HP**[**x,y**] , and **HPxy** are equivalent for general digraphs, from a complexity point of view, and, moreover, they are $\mathcal{NP}$-complete [39]. We now restrict ourselves to tournaments, and as we shall see (from sequential point of view), the first three problems are easy, the fourth is not too complicated either, but the last problem, even though it is polynomial time solvable, is quite complicated. Moreover, a generalization of the **HPxy** problem is $\mathcal{NP}$-complete even for tournaments as we shall see below [22].

It is well known that every tournament contains a hamiltonian path (Redei's theorem) and every strong tournament has a hamiltonian cycle (Camion's theorem). Moreover, a tournament T has a hamiltonian path starting from x if and only if every vertex of T can be reached from x [48] (see also Proposition 5.2). The inductive classical proof of Redei's theorem gives at once a simple $O(n^2)$ algorithm for the first problem. Since sorting by comparisons corresponds to finding a hamiltonian path in a transitive (i.e. acyclic) tournament, we have an $O(n \log n)$-time algorithm in this case. P. Hell and M. Rosenfeld [38] obtained an algorithm with the same complexity solving the **HP** problem for all tournaments (see, also, [17, 47]). The proof of Moon's theorem (every strongly connected tournament is vertex pancyclic [43]) provides an $O(n^3)$-time algorithm for the **HC** problem. Y. Manoussakis [41] constructed an $O(n^2)$-time algorithm for the **HC** problem. This algorithm is optimal since, as it has been proved in [47], there is no sequential algorithm solving the hamiltonian cycle problem in tournaments in time less than cn^2, where c is a constant.

D. Soroker [48] studied the parallel complexity of the above mentioned problems. He proved the following:

Theorem 4.1 *There are $\mathcal{NC}$-algorithms for the* **HP, HPx** *and* **HC** *problems for tournaments.*

Another $\mathcal{NC}$ -algorithm for the **HP** problem in tournaments has been obtained by J. Naor [45]. The most effective parallel algorithm for the **HP** problem for tournaments is due to A. Bar-Noy and J. Naor [23]. They constructed an algorithm which finds a hamiltonian path in time $O(\log n)$ on an $O(n)$ processor CRCW PRAM. Therefore, the last algorithm has an optimal speed-up with respect to the sequential complexity of the problem. The algorithm by A. Bar-Noy and J. Naor can be implemented by generic techniques in the EREW model in parallel time $O(\log^2 n)$ using $O(n)$ processors. Their algorithm uses R. Cole's optimal $\mathcal{NC}$ algorithm for merge sort [26].

The fastest parallel algorithm for the **HC** problem for tournaments is due to E. Bampis, M. El Haddad, Y. Manoussakis and M. Santha [3]. They found a fast parallel procedure which transforms the **HC** problem for tournaments to the **HP** problem for tournaments in the following sense: Given a hamiltonian path in a tournament as input, the procedure constructs a hamiltonian cycle in each non-trivial strongly connected component. The parallel running time of the procedure is $O(\log n)$ using $O(n^2/\log n)$ processors in the CRCW model and $O(\log n \log\log n)$ using $O(n^2/\log n \log\log n)$ processors in the EREW model. Combining the procedure with the algorithm by A. Bar-Noy and J. Naor, the authors of [3] obtained an algorithm with running time $O(\log n)$ using $O(n^2/\log n)$ processors in the CRCW model. Note that this algorithm achieves an optimal speed-up with respect to the sequential complexity of the problem. In the EREW model the algorithm runs in time $O(\log^2 n)$ and uses $O(n^2/\log n \log\log n)$ processors.

For tournaments the **HP[x,y]** problem was solved by C. Thomassen [49] who obtained a theoretical characterization. It follows from this characterization that the existence of a hamiltonian path between x and y can be checked in time $O(n^2)$. Moreover, the proof of the characterization in [49] provides an $O(n^2)$-algorithm for constructing a hamiltonian path between x and y (if one exists).

J. Bang-Jensen, Y. Manoussakis and C. Thomassen [21] considered the much more difficult **HPxy** problem for semicomplete digraphs. The authors of [21] found a polynomial algorithm for solving the **HPxy** problem based on a number of structural results. The question of the existence of such an algorithm for tournaments was raised by Soroker [48].

Theorem 4.2 *[21] There exists an $O(n^5)$ algorithm to check whether a given semicomplete digraph of order n with specified vertices x, y has a hamiltonian (x, y)-path. Moreover, there is an $O(n^7)$ algorithm for constructing a hamiltonian (x, y)-path (if one exists) in a semicomplete digraph of order n with two distinguished vertices x and y.*

The structure of this algorithm is not complicated – it is based on the classical divide and conquer approach – but the proof of its correctness is highly non-trivial.

Note that if we ask for a longest path between x and y in a tournament, then this problem can also be solved in time $O(n^2)$. This follows from Thomassen's characterization in [49]. However, if we insist that the path should go from x to y, then no polynomial algorithm is known. In particular the algorithm in [21] cannot be easily modified to solve this problem, nor does there seem to be an easy reduction of the longest (x, y)-path problem to the **HPxy** problem.

Conjecture 4.3 *There exists a polynomial algorithm which given a semicomplete digraph T and two distinct vertices x, y of T finds a longest (x, y)-path in T.*

Now consider the following generalization of the **HPxy** problem : Given a digraph D and k arcs $e_1, \ldots, e_k \in A(D)$, decide if D has a hamiltonian cycle containing the arcs $e_1, \ldots, e_k$ (the k-**HCA** problem).

Based on the evidence from Theorem 4.2 the authors of [21] raised the following conjecture, the truth of which in the case $k = 1$ follows from Theorem 4.2 :

Conjecture 4.4 *[21] For each fixed k, the k-**HCA** problem is polynomially solvable for semicomplete digraphs.*

Note that J. Bang-Jensen and C. Thomassen proved [22] that the k-**HCA** problem is $\mathcal{NP}$-complete, even for tournaments, when k is not fixed.

At present no $\mathcal{NC}$ algorithms are known for the **HP[x,y]** or **HPxy** problem for tournaments. For a partial result, see Theorem 5.24.

5 Generalizations of tournaments

The first two results described in this section are not very difficult to prove but they are very useful, and, sometimes, allow to essentially simplify the proofs of more complicated theorems and obtain simpler and faster algorithms.

It was shown in [5] that path-mergeable digraphs are recognizable in time $O(m^3)$. This result is based on the following characterization of path-mergeable digraphs.

Theorem 5.1 *[5] A digraph D is path-mergeable if and only if for every pair of distinct vertices $x, y \in V(D)$ and every pair $P = xx_1x_2...x_sy$, $P' = xy_1y_2...y_ty$, $s, t \geq 1$ of internally disjoint (x, y)-paths in D, there exists either an $i \in \{1, ..., s\}$ such that $x_i \rightarrow y_1$, or a $j \in \{1, ..., t\}$ such that $y_j \rightarrow x_1$. Moreover, if D is path-mergeable then P and P' can be merged into one (x, y)-path P^*, so that vertices from P (respectively, P') remain in the same order as on that path. Furthermore, the merging can be done in $O(s + t)$ steps.*

Path-mergeable digraphs form an important family of digraphs since every LISD or LOSD is path-mergeable as shown in [5]. In fact it is easy to show by induction on the length of the paths that the following stronger statement holds:

Proposition 5.2 *If D is a LISD and P, P' are (x, z)-, (y, z)-paths in D which only have z as a common vertex, then D contains a path P^* with $V(P^*) = V(P) \cup V(P')$ such that P ends in z and starts in either x or y. Furthermore the relative ordering of vertices from P and P' is preserved on P^*.* $\diamond$

Two vertices x and y in a digraph D are called *similar* if they are not adjacent and $N^-(x) = N^-(y)$, $N^+(x) = N^+(y)$. For extended LOSD's we have the following result with the same proof as its analogoue for extended LSD's [9]:

Proposition 5.3 *Let D be an extended LOSD and let $P_1 = xx_1 \dots x_sy$ and $P_2 = xz_1 \dots z_tz$ be internally disjoint paths, possibly with $y = z$.*

If no vertex of $V(P_1) \setminus V(P_2)$ is similar to a vertex of $V(P_2) \setminus V(P_1)$, then the following holds:

1. *D contains a path P starting in x and ending in either y or z such that $V(P) = V(P_1) \cup V(P_2)$.*

2. *Furthermore, on P the relative order of vertices from P_i, $i = 1, 2$ is preserved.*

3. *P can be found in time $O(s + t)$.*

The following two results were obtained in [17, 18].

Theorem 5.4 *A strong locally in-semicomplete digraph has a hamiltonian cycle. There is an $O(m + n \log n)$ algorithm for finding a hamiltonian cycle in a strong locally in-semicomplete digraph.*

Theorem 5.5 *A LISD has a hamiltonian path if and only if it contains an in-branching. Given an in-branching of a LISD D, represented by lists of in-neighbours, one can find a hamiltonian path of D in time $O(n \log n)$.*

Let us dwell a little on the last result in order to illustrate how structural properties related to hamiltonian paths in tournaments extend to LISD's. In the classical classroom exercise proof of the fact that every tournament has a hamiltonian path, one simply observes that if P is a path from u to v in a tournament T and x is a vertex not on P, then either $x{\rightarrow}u$ or $v{\rightarrow}x$ or else there are vertices w, z on P such that w is the predecessor of z on P and $w{\rightarrow}x{\rightarrow}z$, i.e. x can be inserted between w and z on P. It is easy to see that using a binary search approach, we can find the right place to insert x by asking at most $\lceil \log |P| \rceil$ questions about directions of certain arcs with x as one of the endpoints. Now let us consider the more general case when T is a LISD and that x has an arc $x{\rightarrow}z$ to P. Using Proposition 5.2, we get that x can be inserted in P before z. Unfortunately, using that approach we may use $O(|P|)$ questions about the orientation of edges per insertion. Instead we shall see that we can still use a type of binary search to find the right place to insert x asking at most $\lceil \log |P| \rceil$ questions about directions of certain arcs: Let s be the middle vertex of the path $P[u, z]$. If $s{\rightarrow}x$, then it follows from the fact that LISD's are path-mergeable that we can merge the two paths $P[s, z]$ and $s{\rightarrow}x{\rightarrow}z$ into one (s, z)-path P^* and hence x can be inserted in the path $P[s, z]$. If $x{\rightarrow}s$, then it follows from Proposition 5.2 that x can be inserted in the path $P[u, s]$ and finally if x and s are non-adjacent, then it follows again from Proposition 5.2 that x can be inserted after s in the path $P[s, z]$. Hence, in all cases we have found a path of size half the original one to consider. This the key observation for the algorithm in [17] (of course we still need datastructures to handle the paths efficiently, etc)

It is clear that, for LOSD's, one can get the same result just by replacing 'in' by 'out'. Note the following more general result (every LISD and LOSD is mergeable [5]).

Theorem 5.6 *[5] A path-mergeable digraph D is hamiltonian if and only if D is strong and $U[D]$ is 2-connected. There is an $O(nm)$-algorithm for finding a hamiltonian cycle in a hamiltonian path-mergeable digraph D on n vertices and m arcs.*

The problem of deciding whether a path-mergeable digraph has a hamiltonian path seems much harder than that of deciding the existence of a hamiltonian cycle. This is because the path-merging property does not imply anything for paths with only one endvertex in common.

Problem 5.7 *Determine the complexity of the hamiltonian path problem for path-mergeable digraphs.*

Clearly, very often strong connectedness is not sufficient to guarantee the existence of a hamiltonian cycle. One reason for this is that strong connectivity is not enough to ensure the existence of a cycle factor, an obvious necessary condition for the existence of a hamiltonian cycle. It is easy to check, in polynomial time, the existence of a such subgraph in a given (general) digraph and find one, if it exists, (see [32, 34, 35]) using any polynomial maximum matching algorithm (for bipartite graphs). In particular, we can do it in time $O(n^{2.5}/\sqrt{\log n})$ applying the algorithm from [2]. J. Bang-Jensen and G. Gutin [9] used this idea showing the following:

Theorem 5.8 *An extended LSD has a hamiltonian cycle if and only if it is strong and has a cycle factor. Given a spanning cycle subgraph of an extended LSD D, a hamiltonian cycle of D can be found in time $O(n^2)$, where n is the number of vertices in D.*

Theorems analogous to Theorem 5.8 have been obtained for semicomplete bipartite digraphs [31, 37, 42], for extended semicomplete digraphs [35] and for extended LOSD's and LISD's [9].

Below we will illustrate some similarities between the cycle structure in some of these classes of digraphs and at the same time illustrate a very useful technique for solving hamiltonian cycle problems in some classes of digraphs (see e.g. [14]).

Let P be a (u, v)-path on one or more vertices (i.e. possibly $u = v$) in a digraph D and let C be a cycle disjoint from P in D. A *partner* of P on C is an arc $x{\rightarrow}y$ of C (i.e. y is the successor of x on C) with the property that $x{\rightarrow}u$ and $v{\rightarrow}y$ are arcs of D. Note that if P has a partner $x{\rightarrow}y$ on C then D contains the cycle $C[y, x]P[u, v]y$. The following more general (and very useful) result is not difficult to prove:

Theorem 5.9 *[14] Let D be a digraph and let $P = u_1u_2 \dots u_r$ be a path and C a cycle in $D - V(P)$. If there exist indices $1 = j_1 < j_2 < \dots < j_s = r + 1$ such that each of the subpaths $P[u_{j_1}, u_{j_2-1}], P[u_{j_2}, u_{j_3-1}], \dots P[u_{j_{s-1}}, u_{j_s-1}]$ has a partner on C, then D has a cycle C' with $V(C') = V(C) \cup V(P)$. Furthermore, given P and C the cycle C' can be found in time $O(|V(C')| \times |V(P)|)$.*

The following lemma is the key tool for proving Theorem 5.8 and the analogous characterization of hamiltonian semicomplete bipartite digraphs.

Lemma 5.10 *Let C and C' be disjoint cycles in a digraph D which is either semicomplete bipartive or extended locally semicomplete. At least one of the following three posibilities hold:*

1. *Either $C{\Rightarrow}C'$, or $C'{\Rightarrow}C$.*
2. *There exist vertices $u \in V(C)$ and $v \in V(C')$ such that $u{\rightarrow}v^+$, $v{\rightarrow}u^+$, where u^+ (v^+, repectively) denotes the successor of u on C (respectively, v on C').*
3. *If D is semicomplete bipartite, then every arc of C has a partner on C' and if D is extended locally semicomplete, then for every arc $x{\rightarrow}x^+$ on C, either the arc $x{\rightarrow}x^+$ has a partner on C' or each of the vertices x, x^+ have partners on C'.*

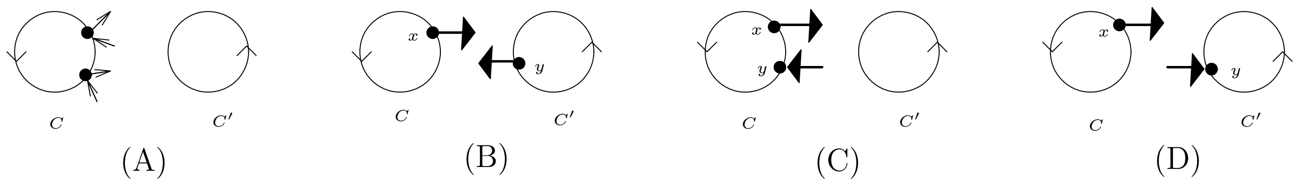

Figure 1: The four possible situations (up to switching the role of the two cycles or reversing all arcs) for arcs between two disjoint cycles in a semicomplete multipartite digraph. In (A) every vertex on C has arcs to and from C'. In (B)-(D) a fat arc indicates that all arcs go in the direction shown from or to the specified vertex (i.e. in (B) all arcs between x and C' leave x).

Note that if C and C' contain vertices x, x' which are similar, then the alternative 2. holds.

Theorem 5.8 does not hold for SMD's as one can see from the examples in [34, 35]. In fact a SMD can be arbitrarily highly connected and have a cycle factor and still not be hamiltonian [13].

Note that if C, C' are disjoint cycles in a SMD D, then (up to switching the role of the two cycles) at least one of the following four cases apply (see Figure 1):

(A) Every vertex on C has an arc to and from C'.

(B) There exist vertices $x \in V(C), y \in V(C')$ such that $x \Rightarrow V(C')$ and $y \Rightarrow V(C)$, or $V(C') \Rightarrow x$ and $V(C) \Rightarrow y$.

(C) C contains distinct vertices x, y such that $x \Rightarrow V(C')$ and $V(C') \Rightarrow y$.

(D) C contains a vertex x such that $x \Rightarrow V(C')$ and C' contains a vertex y such that $V(C) \Rightarrow y$.

The following result was proved in [13]:

Theorem 5.11 *If D is a SMD with disjoint cycles C_1, C_2 for which one of the alternatives (A)-(C) above holds, then in time $O(|V(C_1)| \times |V(C_2)|)$ one can find a cycle C in D with $V(C) = V(C_1) \cup V(C_2)$.*

In the last case when only alternative (D) holds A. Yeo [50] proved (as part of a much stronger result on minimal factors in semicomplete multipartite digraphs, see Theorem 5.12 below) that if there are arcs in both directions between C_1 and C_2 then one can still merge the cycles into a cycle C as above in the same time, unless the following holds for $i = 1$ or $i = 2$: there exists a colour class V_χ of D (that is a maximal independent set of vertices in D) such that all arcs $x \rightarrow y$ from C_{3-i} to C_i satisfy that $x^+, y^- \in V_\chi$.

The following is a simplified statement of the main result in [50]:

Theorem 5.12 *Let D be a strongly connected SMD on n vertices with colour classes $V_1, V_2, \dots, V_c$ and let $\mathcal{F}' = C'_1 \cup \dots \cup C'_r$ be a cycle factor in D. In time $O(n^3)$ one can find either a hamiltonian cycle of D or a new cycle factor $\mathcal{F} = C_1 \cup \dots \cup C_s$, $2 \leq s \leq r$ with the following properties:*

1. *For all $1 \leq i < j \leq s$ none of the alternatives (A)-(C) hold (i.e. (D) holds).*
2. *For every cycle C_i, $i \leq s-1$, there is a colour $\chi_R(C_i) \in \{1, 2, ..., c\}$ and for every cycle C_j, $2 \leq j \leq s$ there is a colour $\chi_L(C_j) \in \{1, 2, ..., c\}$ such that for every arc $x \rightarrow y$ with $x \in V(C_j)$, $j > i$ and $y \in V(C_i)$, we have $y^- \in V_{\chi_R(C_i)}$, $x^+ \in V_{\chi_L(C_j)}$ and $\chi_R(C_i) = \chi_L(C_j)$.*

The following results are corollaries of Theorem 5.12.

Theorem 5.13 *[50] If D is a k-strongly-connected SMD with the property that k is at least the size of the largest independent set (colour class) in D, then D has a hamiltonian cycle and such a cycle can be found in time $O(n^3)$ where n is the number of vertices in D.*

This was conjectured by Y. Guo and L. Volkmann (private communication).

Theorem 5.14 *[50] If D is a regular SMD on n vertices, then D is hamiltonian and a hamiltonian cycle in D can be found in time $O(n^3)$.*

The last result was conjectured by C. Q. Zhang in the case of tournaments [51].

Building upon the most general version of Theorem 5.12 in [50] and a number of new technical results, the authors of this paper and A. Yeo managed to prove the following result.

Theorem 5.15 *[16] The* **HC** *problem is in $\mathcal{P}$ for semicomplete multipartite digraphs.*

The complexity of the algorithm is $O(n^7)$ but no attempt was made to optimize the complexity, since already proving the existence of a polynomial algorithm for the problem was a very complicated task [16].

Somewhat surprisingly, the **HP** problem is much easier than the **HC** problem for semicomplete multipartite digraphs. G. Gutin [33] (see also [32, 34]) found simple necessary and sufficient conditions for a SMD to have a hamiltonian path and showed that these conditions implied a polynomial algorithm to solve the **HP** problem for SMD's. The analogous result holds for extended LSD's [9]. We formulate this result for the last family of digraphs.

Theorem 5.16 *A connected extended LSD D has a hamiltonian path if and only if it contains a 1-path-cycle factor. Given a 1-path-cycle factor of D, one can construct a hamiltonian path of D in time $O(n^2)$.*

Note that, by Theorem 5.16, a hamiltonian path in an extended LSD D (if one exists) can be constructed in time $O(n^{2.5}/\sqrt{\log n})$. Indeed, it is easy to see that a digraph H has a 1-path-cycle factor F if and only if the digraph H', obtained from H by adding a new vertex x together with all possible arcs in both directions between x and $V(H)$, has a cycle factor. Hence, the problem for finding a 1-path-cycle factor is easily transformed to that for finding a cycle factor. The last problem was considered above.

It also follows from the main result of [28] that the hamiltonian path problem for SMDs without 2-cycles and at most two vertices in each colour class is polynomial time solvable. Namely J. F Fink and L. Lesniak-Forster [28] showed the following: Let H be any graph obtained from a complete graph by removing the edges a collection of vertex disjoint paths each of length at most two. Let H' be any orientation of H. Then H' has a hamiltonian cycle if and only if H' is unilaterally connected, i.e. for any choice of $x, y \in V(H')$, H' contains a directed path from x to y or oppositely.

Unlike Theorem 5.8, Theorem 5.16 cannot be generalized to extended LISD's as one can see from an example given in [9].

Conjecture 5.17 *The* **HP** *problem is in $\mathcal{P}$ for extended LISD's.*

The **HP[x,y]** problem for semicomplete bipartite and extended semicomplete digraphs was investigated in [20] and [15], respectively. The authors of both papers generalized the characterization by C. Thomassen for tournaments and proved that the problems can be solved in time $O(n^{2.5}/\log n)$.

J. Bang-Jensen, Y. Guo and L. Volkmann [8] solved the **HP[x,y]** problem for locally semicomplete digraphs by giving a complete mathematical characterization and as a consequence of their characterization it follows that the **HP[x,y]** problem is polynomially solvable for locally semicomplete digraphs.

Conjecture 5.18 *The* **HPxy** *problem is in $\mathcal{P}$ for locally semicomplete digraphs.*

Some support for this conjecture is given in [30] where Y. Guo proved that every 4-strongly-connected locally semicomplete digraph D has an (x, y)-hamiltonian path for any choice of distinct vertices $x, y \in V(D)$. The analogoue of this result for semicomplete digraphs was proved by C. Thomassen [49] and this result was used in the proof of Theorem 4.2.

Now let us consider parallel algorithms for the **HP** and **HC** problems in semicomplete bipartite digraphs. The first problem one faces when trying to develop such algorithms is checking the existence of a 1-path-cycle factor or a cycle factor. In the sequential case, the existence of these can be checked by reducing the problem to a bipartite matching problem. So far no $\mathcal{NC}$ algorithm is known for bipartite matching, but the problem is in $\mathcal{RNC}$ [44]. There seems to be no way to avoid the matching algorithm when checking for a 1-path-cycle factor or a cycle factor in a SBD. In fact, it is shown in [7] that the **HC** problem for SBD's is in $\mathcal{NC}$ if and only if the bipartite matching problem is in $\mathcal{NC}$:

Theorem 5.19 *If $\mathcal{A}$ is an $O(r(n))$-time $p(n)$-processor algorithm for the* **HC** *problem for semicomplete bipartite digraphs on n vertices, then the existence of a perfect matching in a bipartite digraph on n vertices can be decided by an $O(r(n) + n^2/p(n))$-time $p(n)$-processor algorithm.*

The following result is obtained in [7].

Theorem 5.20 *There exists an $O(\log n^4)$-time $O(n^2)$ processor CRCW PRAM algorithm to find a hamiltonian cycle in a strongly connected semicomplete bipartite digraph B, provided that a cycle factor is computed in a preprocessing step. Similarly, given a 1-path-cycle factor, computed in a preprocessing step, a hamiltonian path can be found with the same complexity and processor requirements.*

This algorithm uses the optimal parallel algorithm for the **HP** problem in tournaments as well as a number of fundamental algorithms in parallel computing, such as maximal matching, tree contraction, etc.

In [6] it was pointed out that the algorithms of Theorem 5.20 actually applies to a much more general class of digraphs than just semicomplete bipartite digraphs:

Theorem 5.21 *[6] The algorithms of Theorem 5.20 can be used to solve the hamiltonian path and cycle problems respectively within the same time and processor bounds for any class of digraphs $\mathcal{D}$ with the following properties:*

1. *For every $D \in \mathcal{D}$ and every cycle factor $\mathcal{C}$ of D, the digraph $D_{\mathcal{C}}$ obtained from D by contracting each cycle of $\mathcal{C}$ into one vertex is semicomplete.*
2. *For every $D \in \mathcal{D}$ and disjoint cycles C, C' in D such that there are arcs in both directions between C and C', one can find a cycle C'' such that $V(C'') = V(C) \cup V(C')$ in time $O(\log n)$ using $O(n^2)$ processors.*
3. *If $D \in \mathcal{D}$ and $C^* = c_1 c_2 \dots c_r c_1$ is a cycle in $D_{\mathcal{C}}$ such that no two consecutive arcs on C^* both are contained in a 2-cycle in $D_{\mathcal{C}}$, then one can find a cycle C in B with $V(C) = V(C_1) \cup \dots \cup V(C_r)$ in $O(1)$ time using $O(n)$ processors.*

The following results are a consequence of Theorem 5.21.

Theorem 5.22 *[6] A hamiltonian cycle in an extended semicomplete digraph D can be found:*

- *in $O(\log^4 n)$ time with $O(n^{5.5})$ CRCW processors by a randomized algorithm, and*
- *in $O(\log^4 n)$ time with $O(n^2)$ processors by a deterministic algorithm if a factor $\mathcal{C}$ of D is already found in a preprocessing step.*

Theorem 5.23 *[6] The existence of a hamiltonian path in an extended semicomplete digraph can be decided and a hamiltonian path found, if one exists, within the same complexity and processor bounds as in Theorem 5.22.*

In [6] it was also shown that the **HP[x,y]** problem is solvable efficiently in parallel for SBDs and extended semicomplete digraphs (a path with a cofactor in a digraph D is a path P such that $D - P$ has a cycle factor):

Theorem 5.24 *Given a digraph D which is either an extended tournamnent or semicomplete bipartite and given distinct vertices x and y of D, the existence of a hamiltonian path with endvertices in the set $\{x, y\}$ can be decided and a path found, if one exists:*

- *in $O(\log^4 n)$ time with $O(n^{5.5})$ CRCW processors by a randomized algorithm, and*
- *in $O(\log^4 n)$ time with $O(n^2)$ processors by a deterministic algorithm if, in a preprocessing step, we have decided the existence of an (x, y)-path P with a cofactor $\mathcal{C}$ and a (y, x)-path P' with a cofactor $\mathcal{C}'$ and have found the above paths P and P' and cofactors $\mathcal{C}$ and $\mathcal{C}'$ if such exist.*

6 Totally Φ-decomposable digraphs

Let Φ be a set of digraphs containing the digraph with one vertex. A digraph D is called Φ-*decomposable* if either D has only one vertex or there is a decomposition $D = H[S_1, ..., S_h]$, $h \geq 2$ such that $H \in \Phi$ (we call this decomposition a Φ-*decomposition*). Note that every Φ-graph is Φ-decomposable: just take each S_i as the graph with one vertex.

A digraph D is called *totally* Φ-*decomposable* if either $D \in \Phi$ or there is a Φ-decomposition $D = H[S_1, ..., S_h]$ such that $h \geq 2$, and each S_i is Φ-decomposable. In this case, a Φ-decomposition of D, Φ-decompositions $S_i = H_i[S_{i1}, ..., S_{ih_i}]$ of all S_i which have more than one vertex, Φ-decompositions of those of S_{ij} who has more than one vertex, and so on, form a set of digraphs which will be called a *total* Φ-*decomposition* of D.

Φ_0 denotes the union of all semicomplete multipartite, extended locally semicomplete and acyclic digraphs, Φ_1 is the union of all semicomplete bipartite, extended locally semicomplete and acyclic digraphs. Let Ψ be the union of semicomplete digraphs and acyclic digraphs.

The following result was proved in [9], (see also [27] for more general decomposition results).

Theorem 6.1 *Given a digraph D, one can check if D is totally Φ_i-decomposable ($i = 0, 1$) and, if it is so, find a total Φ_i-decomposition of D in time $O(nm + n^2)$.*

J. Bang-Jensen and J. Huang [19] showed that quasi-transitive digraphs are totally Ψ-decomposable. Using this result they characterized quasi-transitive digraphs containing hamiltonian cycles and hamiltonian paths. The proofs of these characterizations corresponding problems use the analogues of Theorems 5.8 and 5.16 for extended semicomplete digraphs. J. Bang-Jensen and J. Huang [19] noted that their characterizations do not seem to imply polynomial algorithms for the **HP** and **HC** problems for quasi-transitive digraph and conjectured that there exist such algorithms. In [36], G.

Gutin described $O(n^4/\log n)$ algorithms for finding a hamiltonian cycle (path, respectively) in a quasi-transitive digraph D (if D has one). The algorithms are based on an approach which will be applied bellow to get new results for the much more general totally Φ_i-decomposable digraphs ($i = 0, 1$).

We need a simple but important fact on flows in networks. The terminology of flows in networks is rather standard, the undefined terms can be found in [1, 24, 46]. A *circulation* is a flow with value 0. A circulation is a *cycle flow* if the digraph induced by arcs with non-zero flow is just a directed cycle. We shall consider only *integer valued* flows, i.e. flows f such that $f(a)$ is an non-negative integer for every arc a of the network. The following claim can be proved analogously to Theorem 7.2 in [24] using Euler's theorem. We provide a short sketch of the proof. Let $N = (V, A, s, t)$ be a network with an integer (s, t)-flow f of value k. Let M be the directed multigraph obtained from N by replacing each arc $a \in A$ by $f(a)$ copies of it. Let M' be M with a new vertex v and k arcs from v to s as well as k arcs from t to v. Observe that M' is eulerian. We can find an euler tour T in M' in $O(m')$, where m' is the number of arcs in M'. To extract the desired paths and cycles it suffices to traverse T and then delete the new vertex v from all cycles containing it.

Proposition 6.2 *Let N be a network with source s and sink t. Every (s, t)-flow f of value $k \geq 0$ can be decomposed into k flows of value 1 along (s,t)-paths and a number of cycle flows. Such a decomposition of f can be found in time $O(\sum_{a\in A(N)} f(a))$, where $f(a)$ is the number of units of f along an arc a.*

We shall consider some generalizations of the **HP** and **HPx** problems. First we give a few extra definitions. A k-path-cycle factor F *starts* at a vertex x if one of the paths of F starts at x. The *path-covering number* of a digraph D ($pc(D)$) is the minimum integer k such that D has k-path factor. Similarly $pc_x(D)$ is the minimum number of paths in a path factor that starts in x.

The two new problems are:

- Given a digraph D, find a $pc(D)$-path factor of D (the **PF** problem);
- Given a digraph D and a vertex $x \in V(D)$, find a $pc_x(D)$-path factor of D starting at x (the **PFx** problem).

Note that given a $pc(D)$-path factor F of D we can easily construct k-path factors of D for each $k = pc(D) + 1, \ldots, n$ by deleting some arcs from F.

To prove Theorems 6.3 and 6.5, we use a modification of a method suggested in [36]. It was first proved in [9] that the **HP** problem for totally Φ_0-decomposable digraphs and the **HC** problem for totally the Φ_1-decomposable digraphs are polynomial time solvable. In [9], the complexity obtained for both problems was $O(n^5)$. Here we show that it can be decreased to $O(n^4)$.

Theorem 6.3 *If D is totally Φ_0-decomposable, then the* **PF** *problem for D can be solved in time $O(n^4)$.*

Proof: Let $D = R[H_1, ..., H_r]$, where $R \in \Phi_0$, and $H_1, ..., H_r$ are totally Φ_0-decomposable, be a part of a total Φ_0-decomposition of D. Suppose we have obtained solutions to the **PF** problems for $H_1, \ldots, H_r$.

Consider the following set of digraphs

$$\mathcal{S} = \{R[E_{n_1}, ..., E_{n_r}]: \; pc(H_i) \leq n_i \leq |V(H_i)|, \; i = 1, ..., r\},$$

where E_p is a digraph of order p having no arcs, and the network N_R containing the digraph R and two additional vertices (source and sink): s and t such that s and t are adjacent to every vertex of $V(R)$ and the arcs between s (t, resp.) and R are oriented from s to R (from R to t, resp.). Associate with a vertex i of R the lower and upper bounds $pc(H_i)$ and $|V(H_i)|$. Suppose that N_R admits a flow of value k. Then, by Proposition 6.2, there is a collection M_k of k paths and a number of cycles covering $V(R)$. Since a vertex i of R lies on t_i of these paths and cycles, for some t_i such that $pc(H_i) \leq t_i \leq |V(H_i)|$, we can transform M_k into a k-path-cycle factor $F(M_k)$ of a digraph $Q = R[E_{t_1}, \ldots, E_{t_r}]$ such that $Q \in \mathcal{S}$ by replacing the vertex i by t_i independent new vertices such that each new vertex corresponds to one of the occurrences of i in M_k. Since Q is a Φ_0-graph, one can transform, in time $O(n^2)$, $F(M_k)$ into a k-path factor $F'(M_k)$ of Q. Indeed, if Q is acyclic this is trivial. If Q is semicomplete multipartite or extended locally semicomplete, then this follows from Theorem 5.16 and its analogue for semicomplete multipartite digraphs. Finally change $F'(M_k)$ to a k-path factor $F''(M_k)$ of D, by replacing the vertices of each E_{t_i} by t_i paths that form a t_i-path factor of H_i.

Conversely, suppose L_k is a k-path factor of D. For each H_i, $A(H_i) \cap A(L_k)$ induce a collection of α_i vertex disjoint paths in H_i. Clearly $pc(H_i) \leq \alpha_i \leq |V(H_i)|$. Let $Q = R[E_{\alpha_1}, \ldots, E_{\alpha_r}] \in \mathcal{S}$. Then $Q(L_k)$ has a k-path factor which can be obtained from L_k by contracting, for all i, each of the α_i subpaths in H_i to a vertex. It is easy to check that if a digraph from $\mathcal{S}$ has k-path factor, then N_R admits a flow of value k. Hence, the value of a minimum flow in N_R is the path-covering number of D, and, given $pc(H_i)$, $|V(H_i)|$, $i = 1, ..., r$ one can find a $pc(D)$-path factor of D in time $O(n^3)$ (i.e. the time it takes to compute a minimum flow in N_R). This fact leads to an $O(n^4)$ recursive algorithm for finding a $pc(D)$-path factor F of D. $\diamond$

Lemma 6.4 *Let D be either a SBD or an extended LOSD and $x \in V(D)$. Then D has a hamiltonian path starting at x if and only if D contains a 1-path-cycle factor F of D such that the path of F starts at x, and, for every vertex y of $V(D) - \{x\}$, there is an (x, y)-path in D. Moreover, if D has a hamiltonian path starting at x, then, given a 1-path-cycle factor F of D such that the path of F starts at x, the desired hamiltonian path can be found in time $O(n^2)$.*

Proof: As the necessity is clear, we will only prove the sufficiency. Suppose that $F = P \cup C_1 \cup ... \cup C_t$ is a 1-path-cycle factor of D that consists of a path P starting at x and cycles C_i, $i = 1, ..., t$. Suppose also that every vertex of D is reachable from x. Then, w.l.o.g., there is a vertex of P that dominates a vertex of C_1. Let $P = (x = x_1, x_2, ..., x_p)$, $C_1 = (y_1, y_2, ..., y_q, y_1)$ and $x_k \rightarrow y_s$. We shall show how to find a new path starting in x which contains all the vertices of $V(P) \cup V(C_1)$. Repeating this process yields the desired path.

Assume that D is an extended LOSD. If P has a vertex x_i similar (see the definition of similar vertices before Proposition 5.3) to a vertex y_j in C_1, then $x_i \rightarrow y_{j+1}$, $y_j \rightarrow x_{i+1}$ and $P[x_1, x_i]C[y_{j+1}, y_j]P[x_{i+1}, x_p]$ is a path starting from x and containing all the vertices of $P \cup C_1$. If P has no vertex that is similar to a vertex in C_1, then we can apply Proposition 5.3 to $P[x_k, x_p]$ and $x_k C_1[y_s, y_{s-1}]$ and merge these two paths into a path R starting from x_k and containing all the vertices of $P[x_k, x_p] \cup C_1$. Now, $P[x_1, x_{k-1}]R$ is a path starting at x and containing all the vertices of $P \cup C_1$.

Suppose now that D is semicomplete bipartite. Then we have either $y_{s-1} \rightarrow x_{k+1}$, implying that $P[x_1, x_k]C_1[y_s, y_{s-1}]P[x_{k+1}, x_p]$ is a path starting at x and covering all the vertices of $P \cup C_1$, or $x_{k+1} \rightarrow y_{s-1}$. In the last case, we consider the arc between x_{k+2} and y_{s-2}. If $y_{s-2} \rightarrow x_{k+2}$ we can construct the desired path, otherwise we continue to consider arcs between x_{k+3} and y_{s-3} and so on. If we do not construct the desired path in this way, then we obtain that the last vertex of P dominates a vertex in C_1, say $x_p \rightarrow y_1$. Hence $PC_1[y_1, y_q]$ is the desired path.

Using the process above and breath-first search, one can construct an $O(n^2)$-algorithm for finding the desired hamiltonian path starting at x. $\diamond$

Theorem 6.5 *Let D be a totally Φ_1-decomposable digraph. Then the* **HC** *and* **PFx** *problems for D can be solved in time $O(n^4)$.*

Proof: Let $D = R[H_1, ..., H_r]$, where $R \in \Phi_1$ and $H_1, ..., H_r$ are totally Φ_1-decomposable be part of a total Φ_1-decomposition of D. Consider the set $\mathcal{S}$ of digraphs and the network N_R, both introduced in the proof of Theorem 6.3. Suppose that D is strong and that N_R admits a circulation. Then, analogously to the proof of Theorem 6.3, there is a digraph $Q \in \mathcal{S}$ that is strong and has a cycle factor. Hence, by the analogues of Theorem 5.8 for LOSD's and SBD's, Q has a hamiltonian cycle which can be constructed in time $O(n^2)$ given the cycle factor. This cycle can easily be transformed into a hamiltonian cycle of D using the same arguments as we used in the proof of Theorem 6.3.

Similarly, we can transform a hamiltonian cycle of D into a hamiltonian cycle of some $Q \in \mathcal{S}$ in the same way as we transformed a k-path factor of D into a k-path factor of some $Q \in \mathcal{S}$ in the proof of Theorem 6.3. Clearly the existence of a hamiltonian digraph $Q \in \mathcal{S}$ implies that N_R admits a circulation.

This gives the following algorithm for the **HC** problem: Apply the algorithm mentioned in Theorem 6.3 and find solutions of the **PF** for $H_1, ..., H_r$. Then check if D is strong and N_R admits a circulation. If both of these things hold then construct a hamiltonian cycle of D using a cycle factor of a $\mathcal{S}$-graph. It is easy to verify that the complexity of this algorithm is $O(n^4)$.

Now consider the **PFx** problem and enumerate $H_1, ..., H_r$ such that $x \in H_1$. Slightly modify N_R by associating unit lower and upper bounds with the arc from s to the vertex y of R corresponding to H_1. Also modify $\mathcal{S}$ by replacing $pc(H_1)$ by $pc_x(H_1)$ in the definition of $\mathcal{S}$.

Let $k \geq 2$. It easily follows from the proof of Theorem 6.3 that D has a k-path factor starting at x if and only if (the modified) N_R admits a flow of value k (when going from a flow of value k to a k-path factor, we just merge cycles with a path that does not contain x).

Now consider the case when $k = 1$, i.e. we are checking for a hamiltonian path starting at x in D. We show that D has a hamiltonian path starting at x iff N_R admits a flow of value 1 and every vertex of D can be reached from x. Necessity is clear, so we prove sufficiency. Suppose that N_R admits a flow of value 1 and every vertex of D can be reached from x. Suppose also that D has an arc a from $H_2 \cup ... \cup H_r$ to H_1. Then, there is a digraph Q from the modified $\mathcal{S}$ such that $Q = R[E_{n_1}, \ldots, E_{n_r}]$ has a 1-path-cycle factor F starting at a vertex z of E_{n_1}. By the assumption that x can reach all vertices in D, it follows that z can reach all vertices of $Q - E_{n_1}$. Furthermore, the existence of the arc a implies that z can also reach all vertices of E_{n_1}. Thus we have shown that all vertices of Q can be reached from z. Hence, by Lemma 6.4, F can be transformed into a hamiltonian path of Q starting at z. The last one can easily be transformed into a hamiltonian path of D starting at x. This step is similar to one in the proof of Theorem 6.3. The only difference is that now we have some n_1-path factor starting at x in H_1. Instead of substituting just any path for the vertex z, we use the path starting in x to replace the vertex z.

Suppose now that D has no arcs from $H_2 \cup ... \cup H_r$ to H_1. Then $pc_x(H_1) = 1$ by the definition of (the modified) N_R (note that no flow in N_R can send more than one unit of flow through the vertex y that corresponds to H_1 in N_R). Since N_R admits a flow of value 1, some $Q = R[E_{n_1}, \ldots, E_{n_r}] \in \mathcal{S}$ has a 1-path-cycle factor starting at x. Since every vertex of H_1 dominates every vertex of $H_2 \cup ... \cup H_r$, the subgraph of Q induced by $E_{n_2} \cup ... \cup E_{n_r}$ contains a 1-path-cycle factor. Hence, the subgraph of D induced by $H_2 \cup ... \cup H_r$ has a hamiltonian path. This hamiltonian path and a hamiltonian path of H_1 starting at x form a hamiltonian path of D starting at x.

The observations above lead to the following algorithm. Solve the **PF** problem for $H_2, ..., H_r$ and the **PFx** problem for H_1. Construct the modified N_R and find a minimum flow f in it. If the value k of f is more than 1, then use a simple modification of the algorithm from Theorem 6.3 to construct a k-path factor of D starting at x. If $k = 1$, then check whether every vertex of D can be reached from x. If x cannot reach all vertices, then construct a 2-path factor of D starting at x, by considering a hamiltonian path of D (obtained via the flow in N_R) and cutting that path just before the vertex x. Otherwise, construct a hamiltonian path of D starting at x as indicated in the proof above.

It is easy to verify that our algorithm has complexity $O(n^4)$. $\diamond$

As the **HC** problem for semicomplete multipartite digraphs is polynomial time solvable (Theorem 5.15), we suspect that this is also the case for the **HC** problem for totally Φ_0-decomposable digraphs. However, to establish this result (if it is correct) a new approach seems to be needed.

Conjecture 6.6 *The* **HC** *problem for totally Φ_0-decomposable digraphs is polynomial time solvable.*

References

[1] R. K. Ahuja, T. L. Magnanti and J. B. Orlin, **Network Flows**, Prentice Hall, New Jersey (1993).

[2] H. Alt, N. Blum, K. Melhorn and M. Paul, Computing of maximum cardinality matching in a bipartite graph in time $O(n^{1.5}\sqrt{m/\log n})$. *Inf. Proc. Letters* **37** (1991) 237-240.

[3] E. Bampis, M. El Haddad, Y. Manoussakis and M. Santha, A parallel reduction of hamilton cycle to hamilton path in tournaments. *J. Algorithms* **19** (1995) 432-440.

[4] J. Bang-Jensen, Locally semicomplete digraphs: A generalizations of tournaments. *J. Graph Theory* **14** (1990) 371-390.

[5] J. Bang-Jensen, Digraphs with the path-merging property. *J. Graph Theory* **20** (1995) 255-265.

[6] J. Bang-Jensen and M. El Haddad, Parallel algorithms for hamiltonian paths with specified (unordered) endvertices in semicomplete bipartite digraphs and extended tournaments, Preprint no 8, Department of Mathematics and Computer Science, Odense University (1996), www.imada.ou.dk, submitted for publication.

[7] J. Bang-Jensen, M. El Haddad, Y. Manoussakis and T. Przytycka, Parallel algorithms for the Hamiltonian cycle and Hamiltonian path problems in semicomplete bipartite digraphs. *Algorithmica* **17** (1997) 67-87.

[8] J. Bang-Jensen, Y. Guo and L. Volkmann, Weakly hamiltonian-connected locally semicomplete digraphs, *J. Graph Theory* **21** (1996) 163-172.

[9] J. Bang-Jensen and G. Gutin, Paths and cycles in extended and decomposable digraphs. *Discrete Math.* **164** (1997) 5–19.

[10] J. Bang-Jensen and G. Gutin, Paths, trees and cycles in tournaments, *Congressus Nummerantium* **115** (1996) 131-170.

[11] J. Bang-Jensen and G. Gutin, Generalizations of tournaments: A survey, *J. Graph Theory*, to appear.

[12] J. Bang-Jensen and G. Gutin, Vertex heaviest paths and cycles in quasi-transitive digraphs. *Discrete Math.* **163** (1996) 217–223.

[13] J. Bang-Jensen, G. Gutin and J. Huang, A sufficient condition for a semicomplete multipartite digraph to be Hamiltonian, *Discrete Math.* **161** (1996) 1-12.

[14] J. Bang-Jensen, G. Gutin and H. Li, Sufficient Conditions for a Digraph to be Hamiltonian. *J. Graph Theory* **22** (1996) 181-187.

[15] J. Bang-Jensen, G. Gutin, and J. Huang, Weekly hamiltonian-connected ordinary multipartite tournaments. *Discrete Math.* **138** (1995) 63-74.

[16] J. Bang-Jensen, G. Gutin, and A. Yeo, A polynomial algorithm for the Hamiltonian cycle problem in semicomplete multipartite digraphs, submitted.

[17] J. Bang-Jensen and P. Hell, Fast algorithms for finding hamiltonian paths and cycles in in-tournament digraphs. *Discrete Applied Math.* **41** (1991) 75-79.

[18] J. Bang-Jensen, J. Huang and E. Prisner, In-tournaments. *J. Combin. Theory Ser.B* 59 (1993) 267-287.

[19] J. Bang-Jensen and J. Huang, Quasi-transitive digraphs. *J. Graph Theory* **20** (1995) 141-161.

[20] J. Bang-Jensen and Y. Manoussakis, Weakly hamiltonian-connected vertices in bipartite tournaments. *J. Combin. Theory* **B 63** (1995) 261-280.

[21] J. Bang-Jensen, Y. Manoussakis and C. Thomassen, A polynomial algorithm for hamiltonian-connectedness in semicomplete digraphs. *J. Algorithms* **13** (1992) 114-127.

[22] J. Bang-Jensen and C. Thomassen, A polynomial algorithm for the 2-path problem for semicomplete digraphs. *SIAM J. Discrete Math.* **5** (1992) 366-376.

[23] A. Bar-Noy and J. Naor, Sorting, minimal feedback sets, and hamiltonian paths in tournaments. *SIAM J. Disc. Math.* **3** (1990) 7-20.

[24] R.G. Busaker and T. L. Saaty. **Finite Graphs and Networks**, McGraw-Hill, 1965.

[25] J. A. Bondy and U. R. S. Murty. **Graph Theory with Applications**, MacMillan Press, 1976.

[26] R. Cole, Parallel merge sort, *SIAM J. Comp.* **17** (1988) 770-785.

[27] A. Ehrenfrucht, H. N. Gabow, R. M. McConnell and S. J. Sullivan, An $O(n^2)$ divide and conquer algorithm for the prime tree decomposition of two-structures and modular decomposition of graphs, *J. Algorithms* **16** (1994) 283-294.

[28] J. F. Fink and L. Lesniak-Forster, Graphs for which every unilateral orientation is traceable, *Ars Combinatoria* **9** (1980) 113-118.

[29] A. Ghouilà-Houri, Caractérisation des graphes non orientès dont on peut orienter les arrêtes de maniere à obtenir le graphe d'un relation d'ordre, *C.R.Acad. Sci.Paris* **254** (1962) 1370-1371.

[30] Y. Guo, Strongly hamiltonian-connected locally semicomplete digraphs, *J. Graph Theory* **22** (1996) 65-73.

[31] G. Gutin, A criterion for complete bipartite digraphs to be Hamiltonian. *Vestsi Acad. Navuk BSSR Ser. Fiz.-Mat. Navuk* No. 1 (1984) 99-100 (In Russian).

[32] G. Gutin, Finding a longest path in a complete multipartite digraph. *SIAM J. Discrete Math.* **6** (1993) 270-273.

[33] G. Gutin , A characterization of complete n-partite digraphs that have a Hamiltonian path. *Kibernetica* No. 1 (1988) 107-108 (In Russian).

[34] G. Gutin, Cycles and paths in complete multipartite digraphs, theorems and algorithms: a survey. *J. Graph Theory* **19** (1995) 481-505.

[35] G. Gutin, *Paths and cycles in digraphs.* Ph.D thesis, Tel Aviv Univ., 1993.

[36] G. Gutin, Polynomial algorithms for finding hamiltonian paths and cycles in quasi-transitive digraphs. *Australasian J. Combin.* **10** (1994) 231-236.

[37] R. Häggkvist and Y. Manoussakis, Cycles and paths in bipartite tournaments with spanning configurations. *Combinatorica* **9** (1989) 33-38.

[38] P. Hell and M. Rosenfeld, The complexity of finding generalized paths in tournaments. *J. Algorithms* **4** (1983) 303-309.

[39] R. M. Karp, Reducibility among combinatorial problems, in R. E. Miller and J. W. Thatcher (eds.), **Complexity of Computer Computations**, Plenum Press, New York (1972) 85-103.

[40] R.M. Karp and V. Ramachandran, A survey of parallel algorithms for shared memory machines. *Handbook of Theoretical Computer Science*, Chapter 17, MIT Press (1990) 871-941.

[41] Y. Manoussakis, A linear time algorithm for finding Hamiltonian cycle in tournaments. *Discrete Appl. Math.* **36** (1992) 199-202.

[42] Y. Manoussakis and Z. Tuza, Polynomial algorithms for finding cycles and paths in bipartite tournaments. *SIAM J. Disc. Math.* **3** (1990) 537-543.

[43] J.W. Moon, **Topics on Tournaments**. Holt, Rinehart and Winston, N.Y., 1968.

[44] K. Mulmuley, U.V. Vazirani and V. B. Vazirani, Matching is as easy as matrix inversion, *Combinatorica* **7** (1987) 105-113.

[45] J. Naor, Two parallel algorithms in graph theory. Technical Report CS-86-6, Hebrew University, 1986.

[46] H. Papadimitriou, K. Steiglitz, **Combinatorial Optimization: Algorithms and Complexity**. Prentice-Hall Inc., New Jersey, 1982.

[47] D. Soroker, Fast parallel algorithms for graphs and networks, *Ph.D Thesis*, Rept UCB/CSD87/390, CS Division, University of California, Berkeley, 1987.

[48] D. Soroker, Fast parallel algorithms for finding Hamiltonian paths and cycles in tournaments. *J. Algorithms* **9** (1988) 276-286.

[49] C. Thomassen, Hamiltonian connected tournaments. *J. Combin. Theory* **28** (1980) 142-163.

[50] A. Yeo, One-diregular subgraphs in semicomplete multipartite digraphs, *Journal of Graph Theory* **24** (1997) 175-185.

[51] C.-Q. Zhang, Hamilton paths in multipartite oriented graphs. *Annals of Disc. Math.* **41** (1989) 499-514.