2-partitions of digraphs

Jørgen Bang-Jensen

University of Southern Denmark

Gregory celebration, Royal Holloway, January 7, 2017

1Based on joint works with Stephane Bessy, Tilde My Christiansen, Frederic Havet, Nathann Cohen and Anders Yeo
Let $P_1, P_2$ be two (di)graph properties

A $(P_1, P_2)$-partition of a (di)graph $D$ is a 2-partition $(V_1, V_2)$ of $V(D)$ such that $V_1$ induces a (di)graph with property $P_1$ and $V_2$ a (di)graph with property $P_2$.

For example a $(\delta^+ \geq 1, \delta^+ \geq 1)$-partition is a 2-partition of a digraph where each partition induces a subdigraph with minimum out-degree at least 1.

Similarly a (strong,acyclic)-partition is a 2-partition $(V_1, V_2)$ such that $D\langle V_1 \rangle$ is strongly connected and $D\langle V_2 \rangle$ is an avyclic digraph.
Let $P_1, P_2$ be two (di)graph properties.

A $(P_1, P_2)$-partition of a (di)graph $D$ is a 2-partition $(V_1, V_2)$ of $V(D)$ such that $V_1$ induces a (di)graph with property $P_1$ and $V_2$ a (di)graph with property $P_2$.

For example a $(\delta^+ \geq 1, \delta^+ \geq 1)$-partition is a 2-partition of a digraph where each partition induces a subdigraph with minimum out-degree at least 1.

Similarly a (strong,acyclic)-partition is a 2-partition $(V_1, V_2)$ such that $D\langle V_1 \rangle$ is strongly connected and $D\langle V_2 \rangle$ is an avyclic digraph.
Examples for undirected graphs:

- (independent, complete)-partition *split graphs*
- (independent, independent)-partition *bipartite graphs*
- (complete, complete)-partition *complements of bipartite graphs*
To avoid trivial partitions where one vertex on one side is
enough, we consider \([k_1, k_2]\)-partitions, that is, partitions
\((V_1, V_2)\) of \(V\) such that \(|V_1| \geq k_1\) and \(|V_2| \geq k_2\).

For given positive integers \(k_1, k_2\) the \((P_1, P_2)-[k_1, k_2]\)-partition
problem consists in deciding whether a given digraph \(D\) has a
\((P_1, P_2)-[k_1, k_2]\)-partition.

When \(k_1 = k_2 = 1\) we usually just write \((P_1, P_2)\)-partition.
To avoid trivial partitions where one vertex on one side is enough, we consider $[k_1, k_2]$-partitions, that is, partitions $(V_1, V_2)$ of $V$ such that $|V_1| \geq k_1$ and $|V_2| \geq k_2$.

For given positive integers $k_1$, $k_2$ the $(P_1, P_2)$-$[k_1, k_2]$-partition problem consists in deciding whether a given digraph $D$ has a $(P_1, P_2)$-$[k_1, k_2]$-partition.

When $k_1 = k_2 = 1$ we usually just write $(P_1, P_2)$-partition.
Let $\mathcal{H}$ and $\mathcal{E}$ denote the following two sets of natural properties of digraphs all of which can be checked in polynomial time:

\begin{align*}
\mathcal{H} &= \{\text{acyclic, complete, arcless, oriented (no 2-cycle), semicomplete, symmetric, tournament}\} \\
\mathcal{E} &= \{\text{strongly connected, connected, minimum out-degree at least 1, minimum in-degree at least 1, minimum semi-degree at least 1, minimum degree at least 1, having an out-branching, having an in-branching}\}.
\end{align*}

These properties are all \textbf{hereditary}, that is, closed under induced subdigraphs.

These properties are all \textbf{enumerable}, that is, one can enumerate in polynomial time all its inclusion-wise maximal subdigraphs having the property.
Let $\mathcal{H}$ and $\mathcal{E}$ denote the following two sets of natural properties of digraphs all of which can be checked in polynomial time:

\[ \mathcal{H} = \{ \text{acyclic, complete, arcless, oriented (no 2-cycle), semicomplete, symmetric, tournament} \} \]

These properties are all hereditary, that is, closed under induced subdigraphs

\[ \mathcal{E} = \{ \text{strongly connected, connected, minimum out-degree at least 1, minimum in-degree at least 1, minimum semi-degree at least 1, minimum degree at least 1, having an out-branching, having an in-branching} \} \]

These properties are all enumerable, that is, one can enumerate in polynomial time all its inclusion-wise maximal subdigraphs having the property.
### Complexity for arbitrary input digraphs

<table>
<thead>
<tr>
<th>$P_1 \setminus P_2$</th>
<th>strong</th>
<th>conn.</th>
<th>$B^+$</th>
<th>$B^-$</th>
<th>$\delta \geq 1$</th>
<th>$\delta^+ \geq 1$</th>
<th>$\delta^- \geq 1$</th>
<th>$\delta^0 \geq 1$</th>
<th>A</th>
<th>C</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>strong</td>
<td>NPC</td>
<td>NPC</td>
<td>NPC</td>
<td>NPC</td>
<td>NPC</td>
<td>NPC</td>
<td>NPC</td>
<td>NPC</td>
<td>P</td>
<td>P</td>
<td>P</td>
</tr>
<tr>
<td>conn.</td>
<td>NPC$^R$</td>
<td>P</td>
<td>P</td>
<td>P</td>
<td>P</td>
<td>NPC</td>
<td>NPC</td>
<td>NPC</td>
<td>P</td>
<td>P</td>
<td>P</td>
</tr>
<tr>
<td>$B^+$</td>
<td>NPC$^R$</td>
<td>P</td>
<td>NPC</td>
<td>P</td>
<td>NPC</td>
<td>NPC</td>
<td>NPC</td>
<td>NPC</td>
<td>P</td>
<td>P</td>
<td>P</td>
</tr>
<tr>
<td>$B^-$</td>
<td>NPC$^R$</td>
<td>P</td>
<td>NPC</td>
<td>P</td>
<td>P</td>
<td>NPC</td>
<td>NPC</td>
<td>NPC</td>
<td>P</td>
<td>P</td>
<td>P</td>
</tr>
<tr>
<td>$\delta \geq 1$</td>
<td>NPC$^R$</td>
<td>P</td>
<td>P</td>
<td>P</td>
<td>P</td>
<td>NPC</td>
<td>NPC</td>
<td>NPC</td>
<td>P</td>
<td>P</td>
<td>P</td>
</tr>
<tr>
<td>$\delta^+ \geq 1$</td>
<td>NPC$^R$</td>
<td>NPC</td>
<td>NPC</td>
<td>P</td>
<td>NPC</td>
<td>P</td>
<td>NPC</td>
<td>NPC</td>
<td>P</td>
<td>P</td>
<td>P</td>
</tr>
<tr>
<td>$\delta^- \geq 1$</td>
<td>NPC$^R$</td>
<td>NPC</td>
<td>NPC</td>
<td>NPC</td>
<td>NPC</td>
<td>NPC</td>
<td>NPC</td>
<td>NPC</td>
<td>P</td>
<td>P</td>
<td>P</td>
</tr>
<tr>
<td>$\delta^0 \geq 1$</td>
<td>NPC</td>
<td>NPC</td>
<td>NPC</td>
<td>NPC</td>
<td>NPC</td>
<td>NPC</td>
<td>NPC</td>
<td>NPC</td>
<td>P</td>
<td>P</td>
<td>P</td>
</tr>
<tr>
<td>A</td>
<td>P</td>
<td>P</td>
<td>P</td>
<td>P</td>
<td>P</td>
<td>P</td>
<td>P</td>
<td>NPC</td>
<td>P</td>
<td>P</td>
<td>NPC</td>
</tr>
<tr>
<td>C</td>
<td>P</td>
<td>P</td>
<td>P</td>
<td>P</td>
<td>P</td>
<td>P</td>
<td>P</td>
<td>NPC</td>
<td>P</td>
<td>P</td>
<td>P</td>
</tr>
<tr>
<td>X</td>
<td>P</td>
<td>P</td>
<td>P</td>
<td>P</td>
<td>P</td>
<td>P</td>
<td>P</td>
<td>NPC</td>
<td>P</td>
<td>P</td>
<td>P</td>
</tr>
</tbody>
</table>

**Properties:** conn.: connected; $B^+$: out-branchable; $B^-$: in-branchable; A: acyclic; C: complete; X: any property in ‘being independent’, ‘being oriented’, ‘being semi-complete’, ‘being a tournament’ and ‘being symmetric’.

**Complexities:** P: polynomial-time solvable; NPC: NP-complete for all values of $k_1$, $k_2$;
NPC$^L$: NP-complete for $k_1 \geq 2$, and polynomial-time solvable for $k_1 = 1$.
NPC$^R$: NP-complete for $k_2 \geq 2$, and polynomial-time solvable for $k_2 = 1$. 
Theorem

Let $\mathcal{H}$ be a checkable hereditary property, $\mathcal{E}$ be an enumerable property, and let $k_1$ and $k_2$ be two positive integers. One can decide in polynomial time whether a given digraph $D$ has a $(\mathcal{H}, \mathcal{E})-[k_1, k_2]$-partition.

Proof: We shall describe a polynomial-time procedure that for any fixed set $U_1$ of $k_1$ vertices of $D$ decides whether $D$ has an $(\mathcal{H}, \mathcal{E})-[k_1, k_2]$-partition $(V_1, V_2)$ with $U_1 \subseteq V_1$.

Then applying this algorithm to the $O(n^{k_1})$ $k_1$-subsets of $V(D)$, we obtain the desired algorithm.
Theorem

Let $H$ be a checkable hereditary property, $E$ be an enumerable property, and let $k_1$ and $k_2$ be two positive integers. One can decide in polynomial time whether a given digraph $D$ has a $(H, E)$-[$k_1, k_2$]-partition.

Proof: We shall describe a polynomial-time procedure that for any fixed set $U_1$ of $k_1$ vertices of $D$ decides whether $D$ has an $(H, E)$-[$k_1, k_2$]-partition $(V_1, V_2)$ with $U_1 \subseteq V_1$.

Then applying this algorithm to the $O(n^{k_1})$ $k_1$-subsets of $V(D)$, we obtain the desired algorithm.
Theorem

Let $H$ be a checkable hereditary property, $E$ be an enumerable property, and let $k_1$ and $k_2$ be two positive integers. One can decide in polynomial time whether a given digraph $D$ has a $(H, E)-[k_1, k_2]$-partition.

Proof: We shall describe a polynomial-time procedure that for any fixed set $U_1$ of $k_1$ vertices of $D$ decides whether $D$ has an $(H, E)-[k_1, k_2]$-partition $(V_1, V_2)$ with $U_1 \subseteq V_1$.

Then applying this algorithm to the $O(n^{k_1})$ $k_1$-subsets of $V(D)$, we obtain the desired algorithm.
First, we enumerate the maximal subdigraphs of $D - U_1$ with property $E$. This can be done in polynomial time because $E$ is enumerable.

Now for each such subdigraph $F$, (there is a polynomial number of them), we check whether $|F| \geq k_2$ and if $D - F$ has property $H$. This can be done in polynomial time because $H$ is checkable.

In the affirmative, we return ‘Yes’, and in the negative we proceed to the next subdigraph.

If no more subdigraph remains, we return ‘No’.

The above procedure clearly runs in polynomial time.
First, we enumerate the maximal subdigraphs of $D - U_1$ with property $E$. This can be done in polynomial time because $E$ is enumerable.

Now for each such subdigraph $F$, (there is a polynomial number of them), we check whether $|F| \geq k_2$ and if $D - F$ has property $H$. This can be done in polynomial time because $H$ is checkable.

In the affirmative, we return ‘Yes’, and in the negative we proceed to the next subdigraph.

If no more subdigraph remains, we return ‘No’.

The above procedure clearly runs in polynomial time.
First, we enumerate the maximal subdigraphs of $D - U_1$ with property $E$. This can be done in polynomial time because $E$ is enumerable.

Now for each such subdigraph $F$, (there is a polynomial number of them), we check whether $|F| \geq k_2$ and if $D - F$ has property $H$. This can be done in polynomial time because $H$ is checkable.

In the affirmative, we return ‘Yes’, and in the negative we proceed to the next subdigraph.

If no more subdigraph remains, we return ‘No’.

The above procedure clearly runs in polynomial time.
First, we enumerate the maximal subdigraphs of $D - U_1$ with property $E$. This can be done in polynomial time because $E$ is enumerable.

Now for each such subdigraph $F$, (there is a polynomial number of them), we check whether $|F| \geq k_2$ and if $D - F$ has property $H$. This can be done in polynomial time because $H$ is checkable.

In the affirmative, we return ‘Yes’, and in the negative we proceed to the next subdigraph.

If no more subdigraph remains, we return ‘No’.

The above procedure clearly runs in polynomial time.
First, we enumerate the maximal subdigraphs of $D - U_1$ with property $E$. This can be done in polynomial time because $E$ is enumerable.

Now for each such subdigraph $F$, (there is a polynomial number of them), we check whether $|F| \geq k_2$ and if $D - F$ has property $H$. This can be done in polynomial time because $H$ is checkable.

In the affirmative, we return ‘Yes’, and in the negative we proceed to the next subdigraph.

If no more subdigraph remains, we return ‘No’.

The above procedure clearly runs in polynomial time.
We need to show that $D$ has an $(\mathbb{H}, E)$-$[k_1, k_2]$-partition $(V_1, V_2)$ with $U_1 \subseteq V_1$ if and only if there is a maximal subdigraph $F$ of $D - U_1$ with property $E$ of order at least $k_2$ such that $D - F$ has property $\mathbb{H}$.

If there is a maximal subdigraph $F$ of $D - U_1$ with property $E$ of order at least $k_2$ such that $D - F$ has property $\mathbb{H}$, then $(V(D - F), V(F))$ is clearly an $(\mathbb{H}, E)$-$[k_1, k_2]$-partition $(V_1, V_2)$ with $U_1 \subseteq V_1$.

Conversely, assume there is an $(\mathbb{H}, E)$-$[k_1, k_2]$-partition $(V_1, V_2)$ with $U_1 \subseteq V_1$. Then $D\langle V_2 \rangle$ has property $E$ and thus is contained in a maximal subdigraph $F$ of $D - U_1$ with property $E$. Since $F$ is a superdigraph of $D\langle V_2 \rangle$ it has order at least $k_2$. In addition, $U_1 \subseteq V(D - F) \subseteq V_1$, so $D - F$ has the property $\mathbb{H}$, because this property is hereditary and $V_1$ has it.
We need to show that $D$ has an $(\mathbb{H}, \mathbb{E})$-$[k_1, k_2]$-partition $(V_1, V_2)$ with $U_1 \subseteq V_1$ if and only if there is a maximal subdigraph $F$ of $D - U_1$ with property $\mathbb{E}$ of order at least $k_2$ such that $D - F$ has property $\mathbb{H}$.

- If there is a maximal subdigraph $F$ of $D - U_1$ with property $\mathbb{E}$ of order at least $k_2$ such that $D - F$ has property $\mathbb{H}$, then $(V(D - F), V(F))$ is clearly an $(\mathbb{H}, \mathbb{E})$-$[k_1, k_2]$-partition $(V_1, V_2)$ with $U_1 \subseteq V_1$.

- Conversely, assume there is an $(\mathbb{H}, \mathbb{E})$-$[k_1, k_2]$-partition $(V_1, V_2)$ with $U_1 \subseteq V_1$. Then $D\langle V_2 \rangle$ has property $\mathbb{E}$ and thus is contained in a maximal subdigraph $F$ of $D - U_1$ with property $\mathbb{E}$. Since $F$ is a superdigraph of $D\langle V_2 \rangle$ it has order at least $k_2$. In addition, $U_1 \subseteq V(D - F) \subseteq V_1$, so $D - F$ has the property $\mathbb{H}$, because this property is hereditary and $V_1$ has it.
We need to show that $D$ has an $(\mathbb{H}, E)$-[$k_1, k_2$]-partition $(V_1, V_2)$ with $U_1 \subseteq V_1$ if and only if there is a maximal subdigraph $F$ of $D - U_1$ with property $E$ of order at least $k_2$ such that $D - F$ has property $\mathbb{H}$.

- If there is a maximal subdigraph $F$ of $D - U_1$ with property $E$ of order at least $k_2$ such that $D - F$ has property $\mathbb{H}$, then $(V(D - F), V(F))$ is clearly an $(\mathbb{H}, E)$-[$k_1, k_2$]-partition $(V_1, V_2)$ with $U_1 \subseteq V_1$.

- Conversely, assume there is an $(\mathbb{H}, E)$-[$k_1, k_2$]-partition $(V_1, V_2)$ with $U_1 \subseteq V_1$. Then $D\langle V_2 \rangle$ has property $E$ and thus is contained in a maximal subdigraph $F$ of $D - U_1$ with property $E$. Since $F$ is a superdigraph of $D\langle V_2 \rangle$ it has order at least $k_2$. In addition, $U_1 \subseteq V(D - F) \subseteq V_1$, so $D - F$ has the property $\mathbb{H}$, because this property is hereditary and $V_1$ has it.
We need to show that $D$ has an $(\mathcal{H}, \mathcal{E})-[k_1, k_2]$-partition $(V_1, V_2)$ with $U_1 \subseteq V_1$ if and only if there is a maximal subdigraph $F$ of $D - U_1$ with property $\mathcal{E}$ of order at least $k_2$ such that $D - F$ has property $\mathcal{H}$.

- If there is a maximal subdigraph $F$ of $D - U_1$ with property $\mathcal{E}$ of order at least $k_2$ such that $D - F$ has property $\mathcal{H}$, then $(V(D - F), V(F))$ is clearly an $(\mathcal{H}, \mathcal{E})-[k_1, k_2]$-partition $(V_1, V_2)$ with $U_1 \subseteq V_1$.

- Conversely, assume there is an $(\mathcal{H}, \mathcal{E})-[k_1, k_2]$-partition $(V_1, V_2)$ with $U_1 \subseteq V_1$. Then $D\langle V_2 \rangle$ has property $\mathcal{E}$ and thus is contained in a maximal subdigraph $F$ of $D - U_1$ with property $\mathcal{E}$. Since $F$ is a superdigraph of $D\langle V_2 \rangle$ it has order at least $k_2$. In addition, $U_1 \subseteq V(D - F) \subseteq V_1$, so $D - F$ has the property $\mathcal{H}$, because this property is hereditary and $V_1$ has it.
One can easily check that the algorithm described in the proof of Theorem 1 runs in time $O(n^{k_1+c})$ for some constant $c$.

A natural question is then to ask whether the problem could be FPT with respect to $(k_1, k_2)$, that is, in time $f(k_1, k_2)n^c$ for some constant $c$ and computable function $f$.

If not, one may ask if it can be solved in FPT time with respect to $k_1$ or $k_2$ only, that is, in time $g(k_i)n^{h(k_3-i)}$ for some computable functions $g$ and $h$. 
One can easily check that the algorithm described in the proof of Theorem 1 runs in time $O(n^{k_1+c})$ for some constant $c$.

A natural question is then to ask whether the problem could be FPT with respect to $(k_1, k_2)$, that is, in time $f(k_1, k_2)n^c$ for some constant $c$ and computable function $f$.

If not, one may ask if it can be solved in FPT time with respect to $k_1$ or $k_2$ only, that is, in time $g(k_i)n^{h(k_3-i)}$ for some computable functions $g$ and $h$. 

Jørgen Bang-Jensen

2-partitions of digraphs
One can easily check that the algorithm described in the proof of Theorem 1 runs in time $O(n^{k_1+c})$ for some constant $c$.

A natural question is then to ask whether the problem could be FPT with respect to $(k_1, k_2)$, that is, in time $f(k_1, k_2)n^c$ for some constant $c$ and computable function $f$.

If not, one may ask if it can be solved in FPT time with respect to $k_1$ or $k_2$ only, that is, in time $g(k_i)n^{h(k_3-i)}$ for some computable functions $g$ and $h$. 
A base digraph for NP-completeness proofs

Figure: A ring digraph

Jørgen Bang-Jensen
2-partitions of digraphs
For $1 \leq j \leq m$, we associate to the $j$th clause $C_j = (\ell_{j,1} \lor \ell_{j,2} \lor \ell_{j,3})$ the set $W_j$ consisting of three vertices of $R(F)$ representing the occurrences of the literals of $C_j$ in $F$.

**Theorem**

Let $F$ be a 3-SAT formula and let $R(F)$ be the corresponding ring digraph. Then the following holds:

- $R(F)$ contains a directed cycle which avoids at least one vertex from each of the sets $W_1, \ldots, W_m$ if and only if $F$ is a ‘Yes’-instance of 3-SAT.

- $R(F)$ contains two disjoint directed cycles $R_1, R_2$, each of which intersects all the sets $W_1, \ldots, W_m$ if and only if $F$ is a ‘Yes’-instance of NAE-3-SAT.
For $1 \leq j \leq m$, we associate to the $j$th clause $C_j = (\ell_{j,1} \vee \ell_{j,2} \vee \ell_{j,3})$ the set $W_j$ consisting of three vertices of $R(\mathcal{F})$ representing the occurrences of the literals of $C_j$ in $\mathcal{F}$.

**Theorem**

Let $\mathcal{F}$ be a 3-SAT formula and let $R(\mathcal{F})$ be the corresponding ring digraph. Then the following holds:

- $R(\mathcal{F})$ contains a directed cycle which avoids at least one vertex from each of the sets $W_1, \ldots, W_m$ if and only if $\mathcal{F}$ is a ‘Yes’-instance of 3-SAT.

- $R(\mathcal{F})$ contains two disjoint directed cycles $R_1, R_2$, each of which intersects all the sets $W_1, \ldots, W_m$ if and only if $\mathcal{F}$ is a ‘Yes’-instance of NAE-3-SAT.
The legend is the same as in the first table, but we have one more complexity type: NPc*: NP-complete for $k_1, k_2 \geq 2$, and polynomial-time solvable for $k_1 = 1$ or $k_2 = 1$. We also emphasize with P, the problems that are polynomial-time solvable on strong digraphs and NP-complete in the general case.
A digraph $D$ is called **$k$-out-critical** if $\delta^+(D) = k$ and no subset of its vertices can be removed without decreasing the minimum out-degree of the resulting digraph.

Let $X \subseteq V$ be a set of vertices in a digraph $D$ with minimum out-degree at least $k$. A set $X' \subseteq V$ is called **$X$-out-critical** if $X \subseteq X'$, $\delta^+(D\langle X'\rangle) \geq k$ and $\delta^+(D\langle X' - Z\rangle) < k$ for every $\emptyset \neq Z \subseteq X' - X$.

A vertex $v \in V(T)$ is said to be **$k$-out-dangerous** if $d^+(v) < 2k - 1$. 
A digraph $D$ is called $k$-out-critical if $\delta^+(D) = k$ and no subset of its vertices can be removed without decreasing the minimum out-degree of the resulting digraph.

Let $X \subseteq V$ be a set of vertices in a digraph $D$ with minimum out-degree at least $k$. A set $X' \subseteq V$ is called $X$-out-critical if $X \subseteq X'$, $\delta^+(D\langle X'\rangle) \geq k$ and $\delta^+(D\langle X' - Z\rangle) < k$ for every $\emptyset \neq Z \subseteq X' - X$.

A vertex $v \in V(T)$ is said to be $k$-out-dangerous if $d^+(v) < 2k - 1$. 

A digraph $D$ is called $k$-out-critical if $\delta^+(D) = k$ and no subset of its vertices can be removed without decreasing the minimum out-degree of the resulting digraph.

Let $X \subseteq V$ be a set of vertices in a digraph $D$ with minimum out-degree at least $k$. A set $X' \subseteq V$ is called $X$-out-critical if $X \subseteq X'$, $\delta^+(D\langle X' \rangle) \geq k$ and $\delta^+(D\langle X' - Z \rangle) < k$ for every $\emptyset \neq Z \subseteq X' - X$.

A vertex $v \in V(T)$ is said to be $k$-out-dangerous if $d^+(v) < 2k - 1$. 
Lemma

Let $k$ be a fixed integer and let $D$ be a semicomplete digraph with minimum out-degree at least $k$. Then the number of $k$-out-dangerous vertices of $D$ is at most $4k - 3$.

Lemma

Let $D$ be a semicomplete digraph such that $\delta^+(D) \geq k$ and let $X \subseteq V(D)$. Then for every $X$-out-critical set $X'$ in $D$ we have $|X'| \leq \frac{k^2 + 3k + 2}{2} + |X|$. In particular every $k$-out-critical set in $D$ has size at most $\frac{k^2 + 3k + 2}{2}$.
Lemma

Let $k$ be a fixed integer and let $D$ be a semicomplete digraph with minimum out-degree at least $k$. Then the number of $k$-out-dangerous vertices of $D$ is at most $4k - 3$.

Lemma

Let $D$ be a semicomplete digraph such that $\delta^+(D) \geq k$ and let $X \subseteq V(D)$. Then for every $X$-out-critical set $X'$ in $D$ we have $|X'| \leq \frac{k^2 + 3k + 2}{2} + |X|$. In particular every $k$-out-critical set in $D$ has size at most $\frac{k^2 + 3k + 2}{2}$. 
Theorem

For every fixed integer \( k \) there exists a polynomial algorithm that either constructs a \((\delta^+ \geq k, \delta^+ \geq k)\)-partition of a given semicomplete digraph \( D \) or correctly outputs that none exists.

Proof:
It suffices to prove that we can test, for a given partition \((O_1, O_2)\) of the out-dangerous vertices, whether there is a solution with \( O_i \subseteq V_i \).
Theorem

For every fixed integer $k$ there exists a polynomial algorithm that either constructs a $(\delta^+ \geq k, \delta^+ \geq k)$-partition of a given semicomplete digraph $D$ or correctly outputs that none exists.

Proof:
It suffices to prove that we can test, for a given partition $(O_1, O_2)$ of the out-dangerous vertices, whether there is a solution with $O_i \subseteq V_i$. 
Let $X$ be an $O_1$-out-critical set such that $X \subseteq V - O_2$. If no such $X$ exists, there is no solution with $O_i \subseteq V_i$.

Starting from the partition $(V_1, V_2) = (X, V - X)$, and moving one vertex at a time, move vertices of $V_2 - O_2$ which have $d^+_T(V_2)(v) < k$ to $V_1$.

If, at any time, this results in a vertex $v \in O_2$ having $d^+_T(V_2)(v) < k$, or $V_2 = \emptyset$, then there is no good partition with $O_i \subseteq V_i$, $i = 1, 2$ and the algorithm terminates.

Otherwise the algorithm will terminate with $O_2 \subseteq V_2 \neq \emptyset$ and hence it has found an $(\delta^+ \geq k, \delta^+ \geq k)$-partition $(V_1, V_2)$ with $O_i \subseteq V_i$, $i = 1, 2$. 

Jørgen Bang-Jensen

2-partitions of digraphs
• Let \( X \) be an \( O_1 \)-out-critical set such that \( X \subseteq V - O_2 \). If no such \( X \) exists, there is no solution with \( O_i \subseteq V_i \).

• Starting from the partition \( (V_1, V_2) = (X, V - X) \), and moving one vertex at a time, move vertices of \( V_2 - O_2 \) which have \( d^+_T(V_2)(v) < k \) to \( V_1 \).

• If, at any time, this results in a vertex \( v \in O_2 \) having \( d^+_T(V_2)(v) < k \), or \( V_2 = \emptyset \), then there is no good partition with \( O_i \subseteq V_i \), \( i = 1, 2 \) and the algorithm terminates.

• Otherwise the algorithm will terminate with \( O_2 \subseteq V_2 \neq \emptyset \) and hence it has found an \((\delta^+ \geq k, \delta^+ \geq k)\)-partition \((V_1, V_2)\) with \( O_i \subseteq V_i \), \( i = 1, 2 \).
Let $X$ be an $O_1$-out-critical set such that $X \subseteq V - O_2$. If no such $X$ exists, there is no solution with $O_i \subseteq V_i$.

Starting from the partition $(V_1, V_2) = (X, V - X)$, and moving one vertex at a time, move vertices of $V_2 - O_2$ which have $d^+_{T\langle V_2 \rangle}(v) < k$ to $V_1$.

If, at any time, this results in a vertex $v \in O_2$ having $d^+_{T\langle V_2 \rangle}(v) < k$, or $V_2 = \emptyset$, then there is no good partition with $O_i \subseteq V_i, = 1, 2$ and the algorithm terminates.

Otherwise the algorithm will terminate with $O_2 \subseteq V_2 \neq \emptyset$ and hence it has found an $(\delta^+ \geq k, \delta^+ \geq k)$-partition $(V_1, V_2)$ with $O_i \subseteq V_i, i = 1, 2$. 
Let $X$ be an $O_1$-out-critical set such that $X \subseteq V - O_2$. If no such $X$ exists, there is no solution with $O_i \subseteq V_i$.

Starting from the partition $(V_1, V_2) = (X, V - X)$, and moving one vertex at a time, move vertices of $V_2 - O_2$ which have $d^+_T(V_2)(v) < k$ to $V_1$.

If, at any time, this results in a vertex $v \in O_2$ having $d^+_T(V_2)(v) < k$, or $V_2 = \emptyset$, then there is no good partition with $O_i \subseteq V_i, = 1, 2$ and the algorithm terminates.

Otherwise the algorithm will terminate with $O_2 \subseteq V_2 \neq \emptyset$ and hence it has found an $(\delta^+ \geq k, \delta^+ \geq k)$-partition $(V_1, V_2)$ with $O_i \subseteq V_i, i = 1, 2$. 
The correctness of $B$ follows from the fact that we only move vertices that are not dangerous and each such vertex has at least $2k - 1$ out-neighbours in $D$.

Hence, as the vertex that we move does not have $k$ out-neighbours in $V_2$, it must have at least $k$ out-neighbours in $V_1$, so $\delta^+(D(V_1)) \geq k$ will hold throughout the execution of $B$.

By Lemma 3, the number of out-dangerous vertices is at most $4k - 3$ and hence the number of $(O_1, O_2)$-partitions is at most $2^{4k-3}$ which is a constant when $k$ is fixed. Furthermore, by Lemma 4, the size of every $O_1$-critical set is also bounded by a function of $k$ and hence each $(O_1, O_2)$-partition induces only a polynomial number of $O_1$-critical sets.

Thus we obtain the desired polynomial time algorithm by running the subalgorithm $B$ for all possible partitions $(O_1, O_2)$ of the out-dangerous vertices and all possible $O_1$-critical sets. □
The correctness of $B$ follows from the fact that we only move vertices that are not dangerous and each such vertex has at least $2k - 1$ out-neighbours in $D$.

Hence, as the vertex that we move does not have $k$ out-neighbours in $V_2$, it must have at least $k$ out-neighbours in $V_1$, so $\delta^+(D\langle V_1 \rangle) \geq k$ will hold throughout the execution of $B$.

By Lemma 3, the number of out-dangerous vertices is at most $4k - 3$ and hence the number of $(O_1, O_2)$-partitions is at most $2^{4k-3}$ which is a constant when $k$ is fixed. Furthermore, by Lemma 4, the size of every $O_1$-critical set is also bounded by a function of $k$ and hence each $(O_1, O_2)$-partition induces only a polynomial number of $O_1$-critical sets.

Thus we obtain the desired polynomial time algorithm by running the subalgorithm $B$ for all possible partitions $(O_1, O_2)$ of the out-dangerous vertices and all possible $O_1$-critical sets. \qed
The correctness of $B$ follows from the fact that we only move vertices that are not dangerous and each such vertex has at least $2k - 1$ out-neighbours in $D$.

Hence, as the vertex that we move does not have $k$ out-neighbours in $V_2$, it must have at least $k$ out-neighbours in $V_1$, so $\delta^+(D(V_1)) \geq k$ will hold throughout the execution of $B$.

By Lemma 3, the number of out-dangerous vertices is at most $4k - 3$ and hence the number of $(O_1, O_2)$-partitions is at most $2^{4k-3}$ which is a constant when $k$ is fixed. Furthermore, by Lemma 4, the size of every $O_1$-critical set is also bounded by a function of $k$ and hence each $(O_1, O_2)$-partition induces only a polynomial number of $O_1$-critical sets.

Thus we obtain the desired polynomial time algorithm by running the subalgorithm $B$ for all possible partitions $(O_1, O_2)$ of the out-dangerous vertices and all possible $O_1$-critical sets. □
The correctness of $\mathcal{B}$ follows from the fact that we only move vertices that are not dangerous and each such vertex has at least $2k - 1$ out-neighbours in $D$.

Hence, as the vertex that we move does not have $k$ out-neighbours in $V_2$, it must have at least $k$ out-neighbours in $V_1$, so $\delta^+(D\langle V_1 \rangle) \geq k$ will hold throughout the execution of $\mathcal{B}$.

By Lemma 3, the number of out-dangerous vertices is at most $4k - 3$ and hence the number of $(O_1, O_2)$-partitions is at most $2^{4k-3}$ which is a constant when $k$ is fixed. Furthermore, by Lemma 4, the size of every $O_1$-critical set is also bounded by a function of $k$ and hence each $(O_1, O_2)$-partition induces only a polynomial number of $O_1$-critical sets.

Thus we obtain the desired polynomial time algorithm by running the subalgorithm $\mathcal{B}$ for all possible partitions $(O_1, O_2)$ of the out-dangerous vertices and all possible $O_1$-critical sets. $\blacksquare$
The correctness of $B$ follows from the fact that we only move vertices that are not dangerous and each such vertex has at least $2k – 1$ out-neighbours in $D$.

Hence, as the vertex that we move does not have $k$ out-neighbours in $V_2$, it must have at least $k$ out-neighbours in $V_1$, so $\delta^+(D\langle V_1 \rangle) \geq k$ will hold throughout the execution of $B$.

By Lemma 3, the number of out-dangerous vertices is at most $4k – 3$ and hence the number of $(O_1, O_2)$-partitions is at most $2^{4k–3}$ which is a constant when $k$ is fixed. Furthermore, by Lemma 4, the size of every $O_1$-critical set is also bounded by a function of $k$ and hence each $(O_1, O_2)$-partition induces only a polynomial number of $O_1$-critical sets.

Thus we obtain the desired polynomial time algorithm by running the subalgorithm $B$ for all possible partitions $(O_1, O_2)$ of the out-dangerous vertices and all possible $O_1$-critical sets. $\square$
Theorem

The following 2-partition problems are $\mathcal{NP}$-complete for the class of semicomplete digraphs and polynomial for tournaments.

(a) Partitioning into two strong tournaments.
(b) Partitioning into two tournaments both of which have minimum out-degree at least one.
(c) Partitioning into two tournaments so that one has minimum in-degree at least one and the other has minimum out-degree at least one.
Let $D = (V, A)$ be a digraph. For a given 2-partition $(V_1, V_2)$ of $V$ we denote by $B_D(V_1, V_2)$ the spanning bipartite subdigraph induces by the arcs with one end in $V_1$ and the other in $V_2$.

Observation (Alon): For every $k$ there exists a digraph $D$ with minimum out-degree $k$ such that for every 2-partition $(V_1, V_2)$ of $V(D)$ some vertex of $B_D(V_1, V_2)$ has out-degree zero.

This follows from a construction of Thomassen of $k$-out-regular digraphs with no even cycle.
Spanning bipartite digraphs

Let $D = (V, A)$ be a digraph. For a given 2-partition $(V_1, V_2)$ of $V$ we denote by $B_D(V_1, V_2)$ the spanning bipartite subdigraph induces by the arcs with one end in $V_1$ and the other in $V_2$.

**Observation (Alon):** For every $k$ there exists a digraph $D$ with minimum out-degree $k$ such that for every 2-partition $(V_1, V_2)$ of $V(D)$ some vertex of $B_D(V_1, V_2)$ has out-degree zero.

This follows from a construction of Thomassen of $k$-out-regular digraphs with no even cycle.
Let $D = (V, A)$ be a digraph. For a given 2-partition $(V_1, V_2)$ of $V$ we denote by $B_D(V_1, V_2)$ the spanning bipartite subdigraph induces by the arcs with one end in $V_1$ and the other in $V_2$.

**Observation (Alon):** For every $k$ there exists a digraph $D$ with minimum out-degree $k$ such that for every 2-partition $(V_1, V_2)$ of $V(D)$ some vertex of $B_D(V_1, V_2)$ has out-degree zero.

This follows from a construction of Thomassen of $k$-out-regular digraphs with no even cycle.
Spanning bipartite digraphs of minimum out-degree at least 1

**Theorem**

It is polynomial to decide whether a given digraph $D$ has a 2-partition $(V_1, V_2)$ so that $B_D(V_1, V_2)$ has minimum out-degree at least one.

Such a partition exists if and only if every terminal strong component contains an even directed cycle.

**Theorem**

For fixed every choice of natural numbers $k_1, k_2$ such that $k_1 + k_2 \geq 3$ it is NP-complete to decide whether a given digraph $D$ has a 2-partition $(V_1, V_2)$ so that in $B_D(V_1, V_2)$ every vertex of $V_i$ has minimum out-degree at least $k_i$ for $i = 1, 2$. 
Spanning bipartite digraphs of minimum out-degree at least 1

**Theorem**

*It is polynomial to decide whether a given digraph $D$ has a 2-partition $(V_1, V_2)$ so that $B_D(V_1, V_2)$ has minimum out-degree at least one.*

Such a partition exists if and only if every terminal strong component contains an even directed cycle.

**Theorem**

*For fixed every choice of natural numbers $k_1, k_2$ such that $k_1 + k_2 \geq 3$ it is NP-complete to decide whether a given digraph $D$ has a 2-partition $(V_1, V_2)$ so that in $B_D(V_1, V_2)$ every vertex of $V_i$ has minimum out-degree at least $k_i$ for $i = 1, 2$.**
Theorem

For every non-negative integer $K$ there exists an eulerian $K$-strong digraph $D$ such that for every 2-partition $(V_1, V_2)$ the bipartite digraph $B_D(V_1, V_2)$ is not strong.

Theorem

For every non-negative integer $K$ it is NP-complete to decide whether a given $K$-strong eulerian digraph $D$ has a 2-partition $(V_1, V_2)$ such that the bipartite digraph $B_D(V_1, V_2)$ is strong.
Theorem
For every non-negative integer $K$ there exists an eulerian $K$-strong digraph $D$ such that for every 2-partition $(V_1, V_2)$ the bipartite digraph $B_D(V_1, V_2)$ is not strong.

Theorem
For every non-negative integer $K$ it is NP-complete to decide whether a given $K$-strong eulerian digraph $D$ has a 2-partition $(V_1, V_2)$ such that the bipartite digraph $B_D(V_1, V_2)$ is strong.
Let $D$ be a digraph. A $k$-colouring of $V(D)$ is a **k-out-colouring** if no out-neighbourhood is monochromatic.

**Proposition**

For all positive integers $k, r$ there exists a bipartite tournament $B_{k,r}$ with $\delta^+(B_{k,r}) = k$ which has no $r$-out-colouring.

**Theorem**

It is NP-complete to decide whether a bipartite tournament admits a 2-out-colouring.
Let $D$ be a digraph. A $k$-colouring of $V(D)$ is a **k-out-colouring** if no out-neighbourhood is monochromatic.

**Proposition**

For all positive integers $k, r$ there exists a bipartite tournament $B_{k,r}$ with $\delta^+(B_{k,r}) = k$ which has no $r$-out-colouring.

**Theorem**

It is NP-complete to decide whether a bipartite tournament admits a 2-out-colouring.
**Theorem**

Every tournament $T$ with $\delta^+(T) \geq 3$ different from the Paley tournament $P_7$ admits a 2-out-colouring.

**Problem**

Does there exists a function $f(k)$ such that every tournament $T$ with $\delta^+(T) \geq f(k)$ has a 2-partition $(V_1, V_2)$ such that $\delta^+(D(V_i)) \geq k$ for $i = 1, 2$ and $\delta^+(B_D(V_1, V_2)) \geq k$?

Update: YES the function exists (from discussion with Alon).
Theorem

Every tournament $T$ with $\delta^+(T) \geq 3$ different from the Paley tournament $P_7$ admits a 2-out-colouring.

Problem

Does there exists a function $f(k)$ such that every tournament $T$ with $\delta^+(T) \geq f(k)$ has a 2-partition $(V_1, V_2)$ such that $\delta^+(D\langle V_i \rangle) \geq k$ for $i = 1, 2$ and $\delta^+(B_D(V_1, V_2)) \geq k$?

Update: YES the function exists (from discussion with Alon).
Thank you very much for your attention!

* 

Jørgen Bang-Jensen · University of Southern Denmark, Odense

jbj@imada.sdu.dk