Computational Complexity Aspects of Graph Pebbling

Kevin G. Milans (milans@math.illinois.edu)
Joint with Bryan Clark

University of Illinois at Urbana-Champaign

CanaDAM 2009
Montréal, Québec
26 May 2009
Graph Pebbling

A pebbling move removes two pebbles from a vertex and places one on a neighbor.
Graph Pebbling

A pebbling move removes two pebbles from a vertex and places one on a neighbor.

reachability: Given a graph $G$ with pebbles and a target $r$, can we put a pebble on $r$?

In this example: yes

Are there fast algorithms for this problem?

Probably not: many problems are special cases of reachability
Graph Pebbling

- A *pebbling move* removes two pebbles from a vertex and places one on a neighbor.
- **REACHABILITY**: Given a graph $G$ with pebbles and a target $r$, can we put a pebble on $r$?
A *pebbling move* removes two pebbles from a vertex and places one on a neighbor.

**REACHABILITY:** Given a graph $G$ with pebbles and a target $r$, can we put a pebble on $r$?
A pebbling move removes two pebbles from a vertex and places one on a neighbor.

REACHABILITY: Given a graph $G$ with pebbles and a target $r$, can we put a pebble on $r$?
Graph Pebbling

- A pebbling move removes two pebbles from a vertex and places one on a neighbor.

- **REACHABILITY**: Given a graph $G$ with pebbles and a target $r$, can we put a pebble on $r$?
Graph Pebbling

A pebbling move removes two pebbles from a vertex and places one on a neighbor.

REACHABILITY: Given a graph $G$ with pebbles and a target $r$, can we put a pebble on $r$?

In this example: yes
Graph Pebbling

A *pebbling move* removes two pebbles from a vertex and places one on a neighbor.

REACHABILITY: Given a graph $G$ with pebbles and a target $r$, can we put a pebble on $r$?

In this example: yes

Are there fast algorithms for this problem?
Graph Pebbling

- A *pebbling move* removes two pebbles from a vertex and places one on a neighbor.
- **REACHABILITY**: Given a graph $G$ with pebbles and a target $r$, can we put a pebble on $r$?
  - In this example: yes
- Are there fast algorithms for this problem?
  - Probably not: many problems are special cases of **REACHABILITY**.
Early 1970's: Stephen Cook and Leonid Levin independently made a remarkable discovery about a problem "3sat": For each problem \( L \) in NP, it is possible to quickly convert instances of \( L \) to instances of 3sat. The procedure that translates instances of \( L \) to instances of 3sat is called a reduction. Theorem (Cook; Levin) 3sat is NP-complete.
Early 1970’s: Stephen Cook and Leonid Levin independently made a remarkable discovery about a problem “3SAT”:
Early 1970’s: Stephen Cook and Leonid Levin independently made a remarkable discovery about a problem “3SAT”:

- For each problem $L$ in NP, it is possible to quickly convert instances of $L$ to instances of 3SAT.
Early 1970’s: Stephen Cook and Leonid Levin independently made a remarkable discovery about a problem “3SAT”:

For each problem $L$ in NP, it is possible to quickly convert instances of $L$ to instances of 3SAT.
Early 1970’s: Stephen Cook and Leonid Levin independently made a remarkable discovery about a problem “3SAT”:

- For each problem $L$ in NP, it is possible to quickly convert instances of $L$ to instances of 3SAT.

- The procedure that translates instances of $L$ to instances of 3SAT is called a reduction.
Early 1970’s: Stephen Cook and Leonid Levin independently made a remarkable discovery about a problem “3SAT”:
- For each problem $L$ in NP, it is possible to quickly convert instances of $L$ to instances of $3SAT$.
- The procedure that translates instances of $L$ to instances of $3SAT$ is called a reduction.

**Theorem (Cook; Levin)**

$3SAT$ is NP-complete.
Complexity of REACHABILITY

Fact
reachability is in NP.

Theorem
There is a polynomial time reduction from 3sat to reachability.

Corollary (Hurlbert-Kierstead; Watson; Clark-Milans)
reachability is NP-complete. If there is a polynomial time algorithm for reachability, then P=NP.
Complexity of REACHABILITY

Fact
REACHABILITY is in NP.
**Complexity of REACHABILITY**

Fact
REACHABILITY is in NP.

Theorem
*There is a polynomial time reduction from 3SAT to REACHABILITY.*
Fact
REACHABILITY is in NP.

Theorem
*There is a polynomial time reduction from 3SAT to REACHABILITY.*
Fact
REACHABILITY is in NP.

Theorem
There is a polynomial time reduction from 3SAT to REACHABILITY.

Corollary (Hurlbert-Kierstead; Watson; Clark-Milans)
REACHABILITY is NP-complete. If there is a polynomial time algorithm for REACHABILITY, then P=NP.
3SAT

- \( \land \) means “and”, \( \lor \) means “or”, \( \overline{x} \) means “not \( x \)”
3SAT

- $\land$ means “and”, $\lor$ means “or”, $\overline{x}$ means “not $x$”
- A boolean formula in 3CNF:

$$
\phi = (w \lor x) \land (w \lor \overline{x}) \land (\overline{w} \lor y \lor z) \land (x \lor \overline{y} \lor \overline{z})
$$


3SAT

- $\land$ means “and”, $\lor$ means “or”, $\overline{x}$ means “not $x$”
- A boolean formula in 3CNF:

$$
\phi = (w \lor x) \land (w \lor \overline{x}) \land (\overline{w} \lor y \lor z) \land (x \lor \overline{y} \lor \overline{z})
$$
3SAT

- \( \land \) means “and”, \( \lor \) means “or”, \( \overline{x} \) means “not \( x \)”
- A boolean formula in 3CNF:
  \[
  \phi = (w \lor x) \land (w \lor \overline{x}) \land (\overline{w} \lor y \lor z) \land (x \lor \overline{y} \lor \overline{z})
  \]
- 3SAT: Given a 3CNF formula \( \phi \), is \( \phi \) satisfiable?
3sat

- \(\wedge\) means “and”, \(\lor\) means “or”, \(\overline{x}\) means “not \(x\)”
- A boolean formula in 3CNF:

\[
\phi = (w \lor x) \land (w \lor \overline{x}) \land (\overline{w} \lor y \lor z) \land (x \lor \overline{y} \lor \overline{z})
\]

- 3SAT: Given a 3CNF formula \(\phi\), is \(\phi\) satisfiable?

**Definition**

A 3CNF formula \(\phi\) is *simple* if
3SAT

- $\land$ means “and”, $\lor$ means “or”, $\overline{x}$ means “not $x$”
- A boolean formula in 3CNF:
  \[
  \phi = (w \lor x) \land (w \lor \overline{x}) \land (\overline{w} \lor y \lor z) \land (x \lor \overline{y} \lor \overline{z})
  \]
- **3SAT**: Given a 3CNF formula $\phi$, is $\phi$ satisfiable?

**Definition**
A 3CNF formula $\phi$ is *simple* if

1. each variable appears at most twice in its positive form, and
3SAT

- $\land$ means “and”, $\lor$ means “or”, $\overline{x}$ means “not $x$”
- A boolean formula in 3CNF:
  \[
  \phi = (w \lor x) \land (w \lor \overline{x}) \land (\overline{w} \lor y \lor z) \land (x \lor \overline{y} \lor \overline{z})
  \]
- 3SAT: Given a 3CNF formula $\phi$, is $\phi$ satisfiable?

**Definition**

A 3CNF formula $\phi$ is *simple* if

1. each variable appears at most twice in its positive form, and
2. each variable appears at most once in its negative form.
3SAT

- ∧ means “and”, ∨ means “or”, ¬x means “not x”
- A boolean formula in 3CNF:
  \[ \phi = (w \lor x) \land (w \lor \overline{x}) \land (\overline{w} \lor y \lor z) \land (x \lor \overline{y} \lor \overline{z}) \]
- 3SAT: Given a 3CNF formula \( \phi \), is \( \phi \) satisfiable?

**Definition**
A 3CNF formula \( \phi \) is *simple* if

1. each variable appears at most twice in its positive form, and
2. each variable appears at most once in its negative form.

**Proposition**
There is a polynomial time algorithm to convert a 3CNF formula to an equivalent simple 3CNF formula.
3SAT to REACHABILITY

Step 1. Straightforward.

Step 2. \text{npr}: Given a graph $G$ with pebbles and a target $r$, can we put a pebble on $r$ using each edge at most once?

Step 3. Replace each edge with a "one-use" path.
3Sat to Reachability

3Sat \rightarrow S3Sat \rightarrow \text{Reachability}
3SAT to REACHABILITY

Step 1. Straightforward.

Step 2. Given a graph $G$ with pebbles and a target $r$, can we put a pebble on $r$ using each edge at most once?

Step 3. Replace each edge with a "one-use" path.
3SAT to REACHABILITY

Step 1. Straightforward.

Step 2. npr: Given a graph $G$ with pebbles and a target $r$, can we put a pebble on $r$ using each edge at most once?

Step 3. Replace each edge with a “one-use” path.
**3SAT to REACHABILITY**

- **Step 1.** Straightforward.
**3SAT to REACHABILITY**

- **Step 1.** Straightforward.

- **Step 2.** **NPR:** Given a graph $G$ with pebbles and a target $r$, can we put a pebble on $r$ using each edge at most once?
3sat to reachability

\[ \phi = (w \lor x) \land (w \lor \overline{x}) \land (\overline{w} \lor y \lor z) \land (x \lor \overline{y} \lor \overline{z}) \]
**3SAT to REACHABILITY**

\[ \phi = (w \lor x) \land (w \lor \overline{x}) \land (w \lor y \lor z) \land (x \lor \overline{y} \lor \overline{z}) \]

And Gadget

And Gadget
3sat to reachability

\[ \phi = (w \lor x) \land (w \lor \overline{x}) \land (\overline{w} \lor y \lor z) \land (x \lor \overline{y} \lor \overline{z}) \]

And Gadget

Or Gadget
**3SAT to REACHABILITY**

\[
\phi = (w \lor x) \land (w \lor \overline{x}) \land (\overline{w} \lor y \lor z) \land (x \lor \overline{y} \lor \overline{z})
\]

And Gadget

Or Gadget

Variable Gadget
3SAT to REACHABILITY

\[ \phi = (w \lor x) \land (w \lor \overline{x}) \land (\overline{w} \lor y \lor z) \land (x \lor \overline{y} \lor \overline{z}) \]

\[
\begin{array}{cccccc}
W & X & Y & Z \\
2 & 2 & 2 & 2 \\
0 & 0 & 2 & 0 \\
2 & 2 & 2 & 2 \\
\end{array}
\]
3SAT to REACHABILITY

\[ \phi = (w \lor x) \land (w \lor \overline{x}) \land (\overline{w} \lor y \lor z) \land (x \lor \overline{y} \lor \overline{z}) \]

W \rightarrow X \rightarrow Y \rightarrow Z
3sat to reachability

\[ \phi = (w \lor x) \land (w \lor \overline{x}) \land (\overline{w} \lor y \lor z) \land (x \lor \overline{y} \lor \overline{z}) \]
3SAT to REACHABILITY

\[ \phi = (w \lor x) \land (w \lor \overline{x}) \land (\overline{w} \lor y \lor z) \land (x \lor \overline{y} \lor \overline{z}) \]
3SAT to REACHABILITY

\[ \phi = (w \lor x) \land (w \lor \bar{x}) \land (\bar{w} \lor y \lor z) \land (x \lor \bar{y} \lor \bar{z}) \]
3SAT to REACHABILITY

\[ \phi = (w \lor x) \land (w \lor \overline{x}) \land (\overline{w} \lor y \lor z) \land (x \lor y \lor z) \]

![Diagram showing the transformation from 3SAT to REACHABILITY](image-url)
3SAT to REACHABILITY

\[ \phi = (w \lor x) \land (w \lor \bar{x}) \land (\bar{w} \lor y \lor z) \land (x \lor \bar{y} \lor \bar{z}) \]

Diagram: A graph with nodes labeled W, X, Y, Z, and edges connecting them with labels 0 and 2.
3sat to reachability

\[ \phi = (w \lor x) \land (w \lor \overline{x}) \land (\overline{w} \lor y \lor z) \land (x \lor \overline{y} \lor \overline{z}) \]
3SAT to REACHABILITY

\[ \phi = (w \lor x) \land (w \lor \overline{x}) \land (\overline{w} \lor y \lor z) \land (x \lor \overline{y} \lor \overline{z}) \]
3SAT to REACHABILITY

- **Step 1.** Straightforward.
- **Step 2.** **NPR:** Given a graph $G$ with pebbles and a target $r$, can we put a pebble on $r$ using each edge at most once?
3SAT to REACHABILITY

- Step 1. Straightforward.
- Step 2. NPR: Given a graph $G$ with pebbles and a target $r$, can we put a pebble on $r$ using each edge at most once?
- Step 3. Replace each edge with a “one-use” path.
3SAT to REACHABILITY

- Step 1. Straightforward.
- Step 2. NPR: Given a graph $G$ with pebbles and a target $r$, can we put a pebble on $r$ using each edge at most once?
- Step 3. Replace each edge with a “one-use” path.
3SAT to REACHABILITY

- Step 1. Straightforward.
- Step 2. **NPR**: Given a graph $G$ with pebbles and a target $r$, can we put a pebble on $r$ using each edge at most once?
- Step 3. Replace each edge with a “one-use” path.
3SAT to REACHABILITY

Step 1. Straightforward.
Step 2. NPR: Given a graph $G$ with pebbles and a target $r$, can we put a pebble on $r$ using each edge at most once?
Step 3. Replace each edge with a “one-use” path.

Theorem
REACHABILITY is NP-complete even for bipartite graphs with $\Delta(G) \leq 3$ and at most 2 pebbles on each vertex.
Pebbling Number

A distribution of pebbles is solvable if every vertex is reachable.
A distribution of pebbles is solvable if every vertex is reachable.

\[ \pi(G) \]: min \( k \) such that each dist. of \( k \) pebbles is solvable.
Pebbling Number

- A distribution of pebbles is **solvable** if every vertex is reachable.
- $\pi(G)$: $\min k$ such that each dist. of $k$ pebbles is solvable.
- **PEBBLING-NUMBER**: given $G$ and $k$, is $\pi(G) \leq k$?
Beyond NP

- We usually think of NP as containing “hard problems”.

![Diagram showing relationships between P, NP, and NPC classes]
We usually think of NP as containing “hard problems”. It’s all relative: some problems make NP look easy.
Beyond NP

- We usually think of NP as containing “hard problems”.
- It’s all relative: some problems make NP look easy.
- $\Pi_2^P$: class of problems containing NP.
Beyond NP

- We usually think of NP as containing “hard problems”.
- It’s all relative: some problems make NP look easy.
- $\Pi_2^P$: class of problems containing NP.
- Roughly: $P$ is to NP as NP is to $\Pi_2^P$. 
Beyond NP

- We usually think of NP as containing “hard problems”.
- It’s all relative: some problems make NP look easy.
- $\Pi_2^P$: class of problems containing NP.
- Roughly: P is to NP as NP is to $\Pi_2^P$.
- How difficult can $\Pi_2^P$ problems be?
Beyond NP

- We usually think of NP as containing “hard problems”.
- It’s all relative: some problems make NP look easy.
- \( \Pi^P_2 \): class of problems containing NP.
- Roughly: \( P \) is to NP as NP is to \( \Pi^P_2 \).
- How difficult can \( \Pi^P_2 \) problems be?
- (Probably) can’t solve them quickly, even if you have a magical device that solves problems in NP instantly.
We usually think of NP as containing “hard problems”.

It’s all relative: some problems make NP look easy.

$\Pi^P_2$: class of problems containing NP.

Roughly: P is to NP as NP is to $\Pi^P_2$.

How difficult can $\Pi^P_2$ problems be?

(Probably) can’t solve them quickly, even if you have a magical device that solves problems in NP instantly.

Just like P vs. NP, most believe that NP $\subsetneq \Pi^P_2$. 
Beyond NP

- We usually think of NP as containing “hard problems”.
- It’s all relative: some problems make NP look easy.
- $\Pi^P_2$: class of problems containing NP.
- Roughly: P is to NP as NP is to $\Pi^P_2$.
- How difficult can $\Pi^P_2$ problems be?
- (Probably) can’t solve them quickly, even if you have a magical device that solves problems in NP instantly.
- Just like P vs. NP, most believe that NP $\subsetneq \Pi^P_2$.
- Analogous to 3SAT in NP: $\forall\exists3\text{SAT}$ in $\Pi^P_2$. 
We usually think of NP as containing “hard problems”.

It’s all relative: some problems make NP look easy.

Π₂^P: class of problems containing NP.

Roughly: P is to NP as NP is to Π₂^P.

How difficult can Π₂^P problems be?

(Probably) can’t solve them quickly, even if you have a magical device that solves problems in NP instantly.

Just like P vs. NP, most believe that NP ⊈ Π₂^P.

Analogous to 3SAT in NP: ∀∃3SAT in Π₂^P.
Beyond NP

- We usually think of NP as containing “hard problems”.
- It’s all relative: some problems make NP look easy.
- $\Pi^P_2$: class of problems containing NP.
- Roughly: P is to NP as NP is to $\Pi^P_2$.
- How difficult can $\Pi^P_2$ problems be?
- (Probably) can’t solve them quickly, even if you have a magical device that solves problems in NP instantly.
- Just like P vs. NP, most believe that $NP \subsetneq \Pi^P_2$.
- Analogous to $3SAT$ in NP: $\forall \exists 3SAT$ in $\Pi^P_2$. 
Beyond NP

3SAT example:

\[(w \lor x) \land (w \lor \overline{x}) \land (\overline{w} \lor y \lor z) \land (x \lor \overline{y} \lor \overline{z})\]
Beyond NP

3SAT example:

\[ \exists w \exists y \exists x \exists z \ (w \lor x) \land (w \lor \overline{x}) \land (\overline{w} \lor y \lor z) \land (x \lor \overline{y} \lor \overline{z}) \]
Beyond NP

▶ 3SAT example:

$$\exists w \exists y \exists x \exists z \ (w \lor x) \land (w \lor \overline{x}) \land (\overline{w} \lor y \lor z) \land (x \lor \overline{y} \lor \overline{z})$$

▶ \forall \exists 3SAT example:

$$\forall w \forall y \exists x \exists z \ (w \lor x) \land (w \lor \overline{x}) \land (\overline{w} \lor y \lor z) \land (x \lor \overline{y} \lor \overline{z})$$
Beyond NP

3SAT example:

$$\exists w \exists y \exists x \exists z \quad (w \lor x) \land (w \lor \overline{x}) \land (\overline{w} \lor y \lor z) \land (x \lor \overline{y} \lor \overline{z})$$

∀∃3SAT example:

$$\forall w \forall y \exists x \exists z \quad (w \lor x) \land (w \lor \overline{x}) \land (\overline{w} \lor y \lor z) \land (x \lor \overline{y} \lor \overline{z})$$

∀∃3SAT example is a “no” instance: if $w$ is false, first two clauses are unsatisfiable.
Beyond NP

**Theorem**

*There is a polynomial time reduction from ∀∃3SAT to PEBBLING-NUMBER.*
Beyond NP

**Theorem**

There is a polynomial time reduction from $\forall \exists 3\text{SAT}$ to PEBBLING-NUMBER.

**Corollary**

PEBBLING-NUMBER is $\Pi_2^P$-complete.
Complexity of Pebbling Number Variants

- \( \hat{\pi}(G) \): min. \( k \) such that there is a solvable dist. of size \( k \).
Complexity of Pebbling Number Variants

- $\widehat{\pi}(G)$: min. $k$ such that there is a solvable dist. of size $k$.
- A dist. of pebbles covers a function $f : V(G) \to \mathbb{N}$ if there is a sequence of pebbling moves after which: $\forall v$ at least $f(v)$ pebbles on $v$.
Complexity of Pebbling Number Variants

- $\hat{\pi}(G)$: min. $k$ such that there is a solvable dist. of size $k$.
- A dist. of pebbles covers a function $f : V(G) \to \mathbb{N}$ if there is a sequence of pebbling moves after which: $\forall v$ at least $f(v)$ pebbles on $v$.
- $\gamma_f(G)$: min. $k$ such that every dist. of $k$ pebbles covers $f$.
Complexity of Pebbling Number Variants

- $\hat{\pi}(G)$: min. $k$ such that there is a solvable dist. of size $k$.
- A dist. of pebbles covers a function $f : V(G) \rightarrow \mathbb{N}$ if there is a sequence of pebbling moves after which: $\forall v$ at least $f(v)$ pebbles on $v$.
- $\gamma_f(G)$: min. $k$ such that every dist. of $k$ pebbles covers $f$.
- $\gamma(G)$: special case of $\gamma_f(G)$ where $f$ is the unit distribution.
Complexity of Pebbling Number Variants

- $\widehat{\pi}(G)$: min. $k$ such that there is a solvable dist. of size $k$.
- A dist. of pebbles covers a function $f : V(G) \rightarrow \mathbb{N}$ if there is a sequence of pebbling moves after which: $\forall v$ at least $f(v)$ pebbles on $v$.
- $\gamma_f(G)$: min. $k$ such that every dist. of $k$ pebbles covers $f$.
- $\gamma(G)$: special case of $\gamma_f(G)$ where $f$ is the unit distribution.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Question</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cover Pebbling Number</td>
<td>Is $\gamma(G) \leq k$?</td>
<td>polynomial time(^1)</td>
</tr>
<tr>
<td>Optimal Pebbling Number</td>
<td>Is $\widehat{\pi}(G) \leq k$?</td>
<td>NP-complete</td>
</tr>
<tr>
<td>Pebbling Number</td>
<td>Is $\pi(G) \leq k$?</td>
<td>$\Sigma_2^P$-complete</td>
</tr>
</tbody>
</table>

\(^1\)Vuong–Wyckoff, Sjöstrand
Complexity of Pebbling Number Variants

- $\hat{\pi}(G)$: min. $k$ such that there is a solvable dist. of size $k$.
- A dist. of pebbles covers a function $f : V(G) \rightarrow \mathbb{N}$ if there is a sequence of pebbling moves after which: $\forall v$ at least $f(v)$ pebbles on $v$.
- $\gamma_f(G)$: min. $k$ such that every dist. of $k$ pebbles covers $f$.
- $\gamma(G)$: special case of $\gamma_f(G)$ where $f$ is the unit distribution.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Question</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cover Pebbling Number</td>
<td>Is $\gamma(G) \leq k$?</td>
<td>polynomial time$^1$</td>
</tr>
<tr>
<td>Optimal Pebbling Number</td>
<td>Is $\hat{\pi}(G) \leq k$?</td>
<td>NP-complete</td>
</tr>
<tr>
<td>Pebbling Number</td>
<td>Is $\pi(G) \leq k$?</td>
<td>$\Pi_2^P$-complete</td>
</tr>
</tbody>
</table>

Open Problems

- Recall: always $\pi(G) \geq |V(G)|$.

$^1$Vuong–Wyckoff, Sjöstrand
Complexity of Pebbling Number Variants

- $\hat{\pi}(G)$: min. $k$ such that there is a solvable dist. of size $k$.
- A dist. of pebbles covers a function $f : V(G) \rightarrow \mathbb{N}$ if there is a sequence of pebbling moves after which: $\forall v \at least f(v)$ pebbles on $v$.
- $\gamma_f(G)$: min. $k$ such that every dist. of $k$ pebbles covers $f$.
- $\gamma(G)$: special case of $\gamma_f(G)$ where $f$ is the unit distribution.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Question</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cover Pebbling Number</td>
<td>Is $\gamma(G) \leq k$?</td>
<td>polynomial time$^1$</td>
</tr>
<tr>
<td>Optimal Pebbling Number</td>
<td>Is $\hat{\pi}(G) \leq k$?</td>
<td>NP-complete</td>
</tr>
<tr>
<td>Pebbling Number</td>
<td>Is $\pi(G) \leq k$?</td>
<td>$\Pi^P_2$-complete</td>
</tr>
</tbody>
</table>

Open Problems

- Recall: always $\pi(G) \geq |V(G)|$.
- What is the complexity of deciding whether $\pi(G) = |V(G)|$?

---

$^1$Vuong–Wyckoff, Sjöstrand
Complexity of Pebbling Number Variants

- $\widehat{\pi}(G)$: min. $k$ such that there is a solvable dist. of size $k$.
- A dist. of pebbles covers a function $f : V(G) \rightarrow \mathbb{N}$ if there is a sequence of pebbling moves after which: $\forall v$ at least $f(v)$ pebbles on $v$.
- $\gamma_f(G)$: min. $k$ such that every dist. of $k$ pebbles covers $f$.
- $\gamma(G)$: special case of $\gamma_f(G)$ where $f$ is the unit distribution.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Question</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cover Pebbling Number</td>
<td>$\gamma(G) \leq k?$</td>
<td>polynomial time$^1$</td>
</tr>
<tr>
<td>Optimal Pebbling Number</td>
<td>$\widehat{\pi}(G) \leq k?$</td>
<td>NP-complete</td>
</tr>
<tr>
<td>Pebbling Number</td>
<td>$\pi(G) \leq k?$</td>
<td>$\Pi_2^P$-complete</td>
</tr>
</tbody>
</table>

Open Problems

- Recall: always $\pi(G) \geq |V(G)|$.
- What is the complexity of deciding whether $\pi(G) = |V(G)|$?
- Approximation algorithms for $\pi(G)$.

$^1$Vuong–Wyckoff, Sjöstrand